TRAC-M-TM-17-006
November 2016

Metrics for Systems Thinking in the Human Dimension

U.S. Army TRADOC Analysis Center-Monterey
700 Dyer Road, Room 176
Monterey, California 93943-0692

TRAC-M-TM-17-006
November 2016

Metrics for Systems Thinking in the Human Dimension

LTC Casey Connors
Dr. William T. Scherer
Ryan C. Boyer

This study cost the
Department of Defense approximately
$121,000 expended by TRAC in
Fiscal Years 16-17.
Prepared on 20170103
TRAC Project Code # 060316

U.S. Army TRADOC Analysis Center-Monterey
700 Dyer Road, Room 176
Monterey, California 93943-0692

This page left intentionally blank.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)* 01-11-2016	**2. REPORT TYPE** Technical Memorandum	**3. DATES COVERED *(From - To)*** October 2015 – November 2016

4. TITLE AND SUBTITLE Metrics for Systems Thinking in the Human Dimension	**5a. CONTRACT NUMBER** W9124N-15-P-0019
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) LTC Casey Connors Dr. William Scherer (UVA) Mr. Ryan C. Boyer (UVA)	**5d. PROJECT NUMBER** **060316**
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) TRADOC Research Analysis Center, Monterey, CA University of Virginia, Charlottesville, VA	**8. PERFORMING ORGANIZATION REPORT NUMBER** TRAC-M-TM-17-006
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) TRADOC Analysis Center, Headquarters, Fort Leavenworth, KS	**10. SPONSOR/MONITOR'S ACRONYM(S)**
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A: Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The purpose of this memorandum is to provide documentation of research by the University of Virginia in collaboration with TRADOC Analysis Center, Monterey (TRAC-MTRY). This research is focused on methods of analyzing a large corpus of journal length, or longer, documents to quickly characterize the corpus and to find documents most associated with a target set of text. The project's central purpose is to identifying Systems Thinking within a large corpus of documents and measure the extent to which systems thinking is occurring within the domain specified by the breadth of documents included in the corpus. However, our method of document analysis may be useful in assisting analysts in conducting research literature reviews of large amounts of documents by targeting the most specific documents related to the research the analyst is attempting to conduct. Our method may also have other areas of usefulness in intelligent document searches as well as several other applications. The UVA final report, which is included as an attachment, shows the body of the work conducted.

15. SUBJECT TERMS

Document classification, latent Dirichlet allocation, machine learning, natural language processing, systems approach, systems thinking, text analysis, topic model, unsupervised learning

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** LTC Casey Connors
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	U	68	**19b. TELEPHONE NUMBER *(include area code)*** (831) 656-7575

This page left intentionally blank.

Acknowledgements

We would like to acknowledge the contributions made to this project by LTC Christopher Smith.

This page intentionally left blank.

Table of Contents

This page left intentionally blank.

Introduction

Purpose

The purpose of this memorandum is to provide documentation of research by the University of Virginia in collaboration with TRADOC Analysis Center, Monterey (TRAC-MTRY). The research focuses on methods for exploring a large corpus of journal length, or longer, documents to quickly characterize the corpus and find documents most associated with a target set of text. The project's central purpose is to identify Systems Thinking within a large corpus of documents and measure the extent to which systems thinking is occurring within the domain specified by the breadth of documents included in the corpus. However, our method of document analysis may be useful in assisting analysts in conducting research literature reviews of large amounts of documents by targeting the most specific documents related to the research the analyst is attempting to conduct. Our method may have other areas of usefulness, such as intelligent document searches and several other applications. The UVA final report, which is included as an attachment, shows the body of our work. Included is the documentation for an R Shiny app written by the UVA team, which includes instructions for using the app and for using the output of the Latent Dirichlet Allocation (LDA) topic model generated to visualize the document relationships using the open-source network visualization software, Gephi. Our method can be adapted to a wide range of tools, including several network visualization libraries in R and other open source tools.

Background

The Army Operating Concept (AOC) published in 2014 describes the complex world that we face in the future and an integrated approach to dealing with this complex world. This integrated approach means that, especially in a resource constrained environment, we must analyze all of our decisions in a holistic manner that allows senior leaders to better understand how their decisions may affect or be affected by a wide spectrum of threats and environments in order to win in a complex world. One way to do this is using a systems-of-systems approach that breaks down independent problems and shows how decisions are inter-connected. A core aspect of systems-of-systems approaches is systems thinking, which is a concept that is somewhat ill-defined in the literature and has not been satisfactorily measured. In prior work, Dr. William Scherer and Dr. Peter Whitehead [1], proposed a groundwork for a generalized, core language for expressing a systems approach. Using this foundation, known in their work as Dimensions of Systems Thinking (DST), we show that it is possible to develop metrics to express degrees of "systems thinking" and the use of systems-of-systems approaches in published documents. Further, our research yields useful insights on text analytics approaches to analyzing large corpora of documents.

Methodology

Overview

We present a human-in-the-loop methodology that assists researchers and analysts by characterizing a large set of documents. Particularly, we develop a method that finds those documents in the larger corpus that are most closely related to a target set of documents. In this way, our method allows an analyst to determine which documents have the greatest potential for matching or exhibiting the same topical characteristics of the target set of documents. Our research is focused specifically on discovering those documents with the most potential to contain systems thinking and thereby exhibit the characteristics required of good system-of-systems decision making. However, this methodology could be used for any number of target document sets, making it useful for a large number of applications involving characterization of a large corpora of documents.

Our methodology uses a text analytics approach called topic modeling. Topics are constructed from a corpus of documents using Latent Dirichlet Allocation, described in detail in Appendix B and in our paper currently submitted to a peer-reviewed journal [2]. Each topic is then reviewed by a panel of subject matter experts (SMEs) that determine topic definitions. At least one of these topics should contain the systems thinking topic or the topic most closely associated with the target set of documents that the analyst is interested in. An iterative approach and a minimum number of target documents is necessary to allow a target topic to emerge. Once these topics are defined, a mathematical distance measure is used to discover those papers in the corpus that are most closely related to the topics and to the target set of documents. In our research, we use simple proportions as our distance measure, but any number of mathematical techniques could be explored in future research.

At this point, SMEs can then identify a set of best and worst related documents, examine those documents to understand the corpus, and identify potential systems thinking being used in the corpus (or targeted topic area). Our final report at Appendix B details the entire methodology and results of this research. The draft paper is an adaptation of the final report with some improvements [2]. The documentation for the software developed at UVA can be found at Appendix C. Any request for the code should be sent to TRAC-Monterey

Appendix A – References

[1] N Peter Whitehead, William T Scherer, and Michael C Smith, "Systems Thinking About Systems Thinking A Proposal for a Common Language," *IEEE Systems Journal*, vol. 9, no. 4, pp. 1-12, December 2015.

[2] Ryan Boyer, William T. Scherer, Cody H. Fleming, MAJ Casey Connors, and N. Peter Whitehead, “Using Topic Models To Understand Systems and Identify Strong Systems Thinking” Submitted for peer-reviewed journal, October 2016.

Appendix B - Final Report "Unsupervised Methods to Discover Evidence of Systems Thinking in Corpora of Text Documents"

TRAC Final Report W9124N-15-P-0019

A version of this report has been submitted for a peer-reviewed journal.

Unsupervised Methods to Discover Evidence of Systems Thinking in Corpora of Text Documents

***Abstract*— Systems thinking characterizes the analytical methods needed to effectively design, maintain, and utilize systems; prior work has shown that the there is a language of systems thinking and that it can be quantified within documents using supervised learning methods. Building on this foundation, we present an unsupervised, human-in-the-loop methodology that utilizes topic models to facilitate the identification of systems thinking within a corpus of documents. The methodology creates a topic model of a corpus and uses each document's topic proportion in a systems thinking topic as a proxy measure for the potential of strong systems thinking in the document. The novel aspect of the methodology is in the seeding of the corpus. The user seeds the corpus with several documents that demonstrate strong systems thinking, which encourages the unsupervised topic model to generate a structure aligned with the users' goals of identifying systems thinking. This causes a systems thinking topic to emerge. Though this method is exploratory, not prescriptive like the prior methodology, it requires no grading of documents, which makes it significantly faster. We use a graded corpus to demonstrate the method's effectiveness; a Tukey test reveals that the top echelon of strong systems thinking papers have significantly higher mean topic proportions in the emerging systems thinking topic than lower graded papers. Furthermore, the methodology can be utilized to overview a system, provide research direction, and to find other topics and concepts of interest within a corpus, which we demonstrate through a case study on a corpus of documents related to the human dimension within the Army. Subjectivity is still inherent in the definition of strong systems thinking and in the interpretation of topics, but this is what makes the human-in-the-loop methodology so effective. The topic model structures information in a way that human intuition can handle the subjectivity, and iteration builds confidence.**

***Index Terms*— Document classification, latent Dirichlet allocation, machine learning, natural language processing, systems approach, systems thinking, text analysis, topic model, unsupervised learning**

I. INTRODUCTION

THE ability to objectively identify good systems engineering and systems design would be useful to systems practitioners everywhere, allowing them to see examples of strong engineering and learn from them accordingly. However, systems engineering is still a field without clear boundaries, and good systems design is very subjective. An expert can often identify good design, but it is usually just because the system functions smoothly and effectively, not as a result of the design process. However, there is a language that characterizes systems thinking, that is the systemic, goal-driven, new-eyed thinking that enables practitioners to design innovative solutions and not neglect important details. This “systems approach” should lead to an effective system design with minimal conflicts and minimal unexpected pitfalls.

Prior work has both defined a lexicon of systems thinking and used supervised learning methods to classify papers with good or bad systems thinking with accuracy on the order of 70% [1], [2]. While this supervised method is relatively effective, it requires a substantially graded or tagged dataset to be implemented. This time consuming and subjective process is a huge deterrent for use of this methodology in practice.

We present a human-in-the-loop iterative method of identifying documents with strong potential for systems thinking using topic models, an unsupervised text analytic technique that automatically structures text documents according to their themes. Topic models model a collection of documents, with each document as a mixture of topics, where topics are distributions of words present in all of the documents. By coercing a systems thinking topic to emerge by seeding the unread papers with a small subset of known papers with good systems thinking, other documents can be tagged with a proportion of the document that belongs to this systems thinking topic. This proportion serves a proxy for the potential for systems thinking. This methodology can also be used to survey and visualize a collection of research documents, providing quick information to practitioners to direct their research.

This methodology for identifying systems thinking does not require extensive grading, making it more rapid than the supervised methodology. This also allows for seeds to be changed quickly as goals change. However, this methodology is fundamentally explorative (not predictive). Our methodology is also limited by the subjective nature of what defines good systems thinking and the subjective interpretation of topics within the topic models, though the prior supervised method suffered similar limitations. Nevertheless, these subjective limitations can also be considered strengths of our human-in-the-loop iterative methodology, because people have intuition and expertise to handle abstract concepts more effectively than computers.

After a brief discussion of the mathematics of topic models and language of systems thinking, we describe the human-in-the-loop methodology for utilizing topic models. We then use the graded corpus from prior work to evaluate our methodology. We show that it is effective at finding documents with strong systems thinking, though it does it in an exploratory, not prescriptive, way. Finally, we present a case study demonstrating the utilization of the methodology to analyze and summarize a corpus related to human dimension programs in the military.

II. The Language of Systems Thinking and Measurement of Systems Thinking

The systems approach is a comprehensive method of designing and understanding systems, which looks at how the system functions as a whole and not how its individual pieces function [3].

This systems approach can contrast with modern understandings of systems engineering, which is sometimes seen as only a systematic process that should be walked through for a system to be designed effectively. However, systems thinkers both past and present feel that this is insufficient for good systems design, and that it is the pairing of good processes with holistic views and well-defined goals that generate effective systems [4], [5].

Still, as systems engineering is fundamentally multidisciplinary and interdisciplinary, the language of systems and the systems approach can quickly be confounded. Ackoff established a language to describe systems, systems of systems, and their parts [3]. Whitehead et al. extended Ackoff's linguistic foundation by creating a taxonomy of the systems approach, which they call the "dimensions of systems thinking" [1]. Both argue that clear communication and clear understanding are essential for systems research to thrive. Standing on the foundation of the systems approach, consensus between stakeholders is essential for a systems success; this consensus will be elusive if systems researchers and systems engineers cannot communicate effectively.

Whitehead et al. leveraged the dimensions of systems thinking to measure the quality of systems thinking in technical reports [2]. This work demonstrated a proof of concept that the language of systems thinking can be objectively classified using natural language processing and supervised learning methods. He began by grading a corpus of 295 documents, giving 62 life-cycle analysis papers either "top" for good systems thinking or "bottom" for poor systems thinking and 233 IEEE papers between 1 (poor systems thinking) and 10 (excellent systems thinking). The 233 IEEE papers served as the training set while the 62 life-cycle analysis papers served as the test set. Whitehead et al. used two learners to evaluate the potential for measuring systems thinking. By converting all the documents to vectors via a term frequency-inverse document frequency (tf-idf) matrix, he identified the Rocchio centroid classification vectors between the good systems thinking documents in the training set (7 or higher) and poor systems thinking documents in the training set (4 or lower). This vector space model classified the test set with an accuracy of 61% using cosine similarity between the vectors. Additionally, Whitehead et al. utilized Quadratic Discriminant Analysis on the tf-idf vectors, which gave an accuracy of 68%. These accuracy rates are not perfect, but do demonstrate that systems thinking can be identified computationally. The obvious caveats to this methodology is that grading a training corpus is a time-consuming and subjective process, and that larger training sets are required for larger and more diverse corpora.

III. Latent Dirichlet Allocation

Topic models provide an alternative, unsupervised way of classifying text that can be used to identify documents with strong systems thinking. Topic models are mixture models (a type of generative probabilistic model) that model the themes of documents within a corpus [6]. Latent Dirichlet Allocation (LDA) is one of the most prevalent topic models used today and is utilized in all examples in this paper. LDA models documents as mixtures of topics and topics as probabilistically weighted lists of words. Given a set of topics, a new document can be generated by choosing proportions of these topics and drawing words from these topics according to these proportions [7]. Operating on the bag-of-words model of text, this document simulation portion is not actually used, but the distributions found by training the model provide keen insight to the nature of the corpus and the documents within the corpus.

Mathematically, LDA is most easily understood through plating notation, shown in Fig. 1 with variables and parameters explained in **TABLE I**. In Fig. 1 the circles represent random variables; grey circles are observable and white circles are hidden. The arrows represent dependence of the random variables; for example, θ, the topic distribution of a document, depends on the Dirichlet parameters α. The boxes represent repetition; for example there is one word probability distribution, η, for each topic, where K is the number of topics.

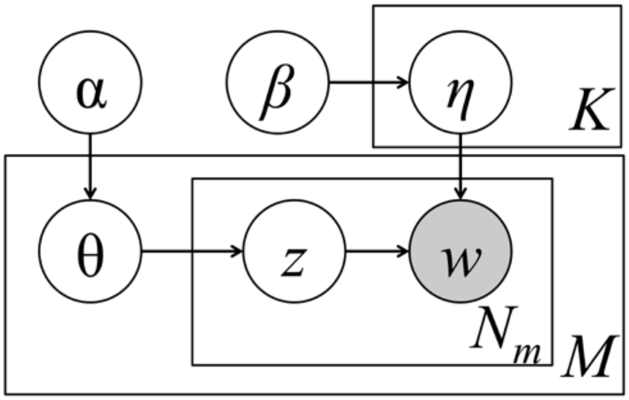


Fig. 1 Smoothed LDA in Plating Notation. Plating notation visualizes the generative model by making the dependencies and repetition of random variables easily visible. Boxes represent repetition and circles represent random variables; shaded circles are observable random variables.

TABLE I
Variables and Parameters for Smoothed LDA

Variable or Parameter	Type	Explanation	Quantity
α	Positive, Real Number vector	Dirichlet Parameter, prior for the per-document topic distributions	K
β	Positive, Real Number vector	Dirichlet Parameter, prior for the per-topic word assignments and distribution	W
θ	Multinomial Distribution, vector of length K with values between 0 and 1 inclusive, summing to 1	Topic proportions for a given document	M
η	Multinomial Distribution, a vector of length K with values between 0 and 1 inclusive, summing to 1	Word probabilities for a given topic	W
z	Integer between 1 and K	Word assignment; assigns a given word from a given document to a specific topic	Total number of words in the corpus, counting duplicates
W	Positive Integer	Number of Unique Words in Corpus	
M	Positive integer	Number of documents in corpus	
N	Positive Integer	Total number of unique words in corpus	
K	Positive Integer	Number of Topics, assigned by user	
i, j, t	Positive Integer	Indexing variables	

Intuitively, repeated sampling of the words within the corpus identifies words that commonly occur together, which is used in defining the topics. In practice, the Bayesian inference for a topic model is complicated. While the total probability of the model can be explicitly written (as show in equation 1**),** the distributions are intractable to calculate exactly. Often variational Bayes, collapsed Gibbs sampling, or expectation propagation are used to approximate these distributions [7], [8], [9]. Furthermore, good choices for the α and β hyper-parameters can be iteratively approximated [10].

$$P(\boldsymbol{W}, \boldsymbol{z}\, \boldsymbol{\theta}, \boldsymbol{\eta} | \alpha, \beta) = \prod_{i=1}^{K} P(\eta_i|\beta) \prod_{j=1}^{M} P(\theta_j|\alpha) \prod_{t=1}^{N} P(z_{j,t}|\theta_j) P\left(W_{j,t}\middle|\eta_{z_{j,t}}\right) \quad (1)$$

LDA does not perfectly reflect the way speakers and writers use language. It treats all documents under the bag-of-words model, which assumes that order of words does not matter. Furthermore, all topics are considered independent and uncorrelated under an LDA model. Yet these simplifications do not hold back its descriptive power.

Topic models are seeing growing utilization. In addition to understanding the themes of a corpus and the documents within it, topic models are being used to understand trends through time, to improve information retrieval processes, and to tag documents appropriately for further sorting or cataloging [11], [12], [13], [14].

There are readily available implementations of LDA in most programming languages. The work in this paper utilized Mallet, an open-source, Java-based package available from University of Massachusetts Amherst [15]. Mallet is one of the most popular implementations today as it is very fast, very light, can be run from the command line, and can be customized extensively. Though we use Mallet and LDA, the methodology that is presented for utilizing topic models to identify systems thinking is not unique to Mallet or LDA.

IV. Human-in-the-Loop Methodology

The human-in-the-loop methodology for using topic models to identify documents with potential for systems thinking is fundamentally iterative and explorative. At a high level, the user seeds the corpus with documents that they believe describe or exhibit strong systems thinking, and then they run a topic model on the new corpus. The topic proportions of documents in the emerging systems thinking topic, which is filled with the dimensions of systems thinking or their synonyms, becomes a proxy measure identifying the strong potential for systems thinking. This idea of seeding an unsupervised method to influence its structure is novel, unique, and effective for data exploration.

The human-in-the-loop methodology can also be used to visualize and understand a system, to identify relations between documents, or to find documents related to other ideas. This report will first walkthrough and discuss the methodology for identifying systems thinking and then briefly discuss its use in exploring a corpus for understanding. **Fig. 2** presents the flowchart for both these processes.

A. Methodology For Identifying Systems Thinking

The user begins by gathering a corpus of unread documents that they are interested in, preferably with over 300 documents of at least a page in length, though smaller corpora may work depending on their content. The user then “seeds” the corpus. He or she adds specific papers to the corpus that are representative of good systems thinking or actively describe good systems. This seeding methodology seeks to guide the topic model. The language within these seed papers will lead to the systems thinking topic output from the topic model. To the authors’ knowledge, the idea of seeding the data to guide an unsupervised learner for directed information has never been published.

Despite Whitehead et al.’s proposal for a common language, the language of systems thinking will differ between domains in practice. Ideally the seed papers will discuss and use systems thinking by utilizing the equivalent dimensions of systems thinking within that domain. However, in our experiments with this methodology, a meta-paper discussing systems thinking from the systems thinking domain has been sufficient.

The number of seed papers required varies due to the number of papers in the corpus of interest and their thematic spread. A good rule of the thumb, however, is one seed paper for every 40 to 60 papers of interest with a minimum of three seeds. For larger corpora, this ratio decreases dramatically. In practice getting the systems thinking topic to emerge can be difficult, and iteration is required.

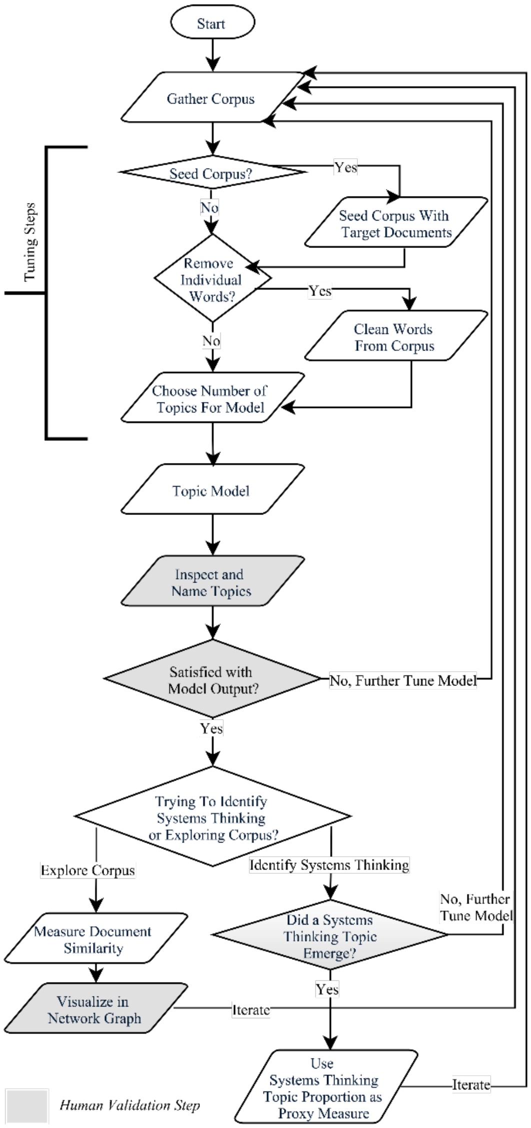

Start
Gather Corpus
Tuning Steps
Seed Corpus?
Yes
No
Seed Corpus With Target Documents
Remove Individual Words?
Yes
No
Clean Words From Corpus
Choose Number of Topics For Model
Topic Model
Inspect and Name Topics
Satisfied with Model Output?
No, Further Tune Model
Yes
Trying To Identify Systems Thinking or Exploring Corpus?
Explore Corpus
Identify Systems Thinking
Measure Document Similarity
Did a Systems Thinking Topic Emerge?
No, Further Tune Model
Yes
Visualize in Network Graph
Iterate
Use Systems Thinking Topic Proportion as Proxy Measure
Iterate
Human Validation Step

Fig. 2 Process flowchart for using a topic model to explore a corpus or identify systems thinking. The human-in-the-loop methodology allows the user to guide an unsupervised learning method (the topic model)

Next, the user then prepares the corpus for topic modeling by cleaning specific words from each document. This cleaning serves to focus the topics from the model on valuable information. Typically the user removes stop words, which are words that have limited meaning on their own, such as common articles, prepositions, adverbs, and transitions words. On additional iterations of the human-in-the-loop methodology, a user might remove context specific words that provide limited information. For example, a corpus focusing on the canine diseases might remove the word “dog” as it will appear in every paper and provide no new information in any topic.

The user picks a number of topics for the model (and any other model parameters if using a non-LDA model) and generates the topic model. The number of topics will tell the algorithm how many topics to look for within the corpus. Fewer topics result in more high-level themes being revealed, at the expense of detail and the possibility of convoluted ideas. More topics result in more detail and potential for specificity in each topic, but at the expense of possibly duplicated topics and the possibility for data overload. Several authors have suggested means of computationally identifying the ideal number of topics for an LDA topic model, but we believe that manual tuning by inspection and repetition provides the desired result more quickly with the added benefit of increased knowledge and understanding of the corpus [16], [17].

The user then inspects the topics produced by the model and gives them appropriate names. This provides insight into the makeup of the corpus and allows the user to see if a system thinking topic emerged. A systems thinking topic would be one where the dominant words in the topic are made of the dimension of systems thinking or their appropriate synonyms in the domain of the corpus.

If a system thinking topic emerged, it can be used as a proxy measure for the potential for systems thinking. The user can begin reading the documents with high proportions of that systems thinking topic to see their analysis and learn from them. If a system thinking topic did not emerge, the user must iterate through the human-in-the-loop methodology again. The topic model should be changed based on the outputs the user saw in the first iteration. For instance, if the topics are all very high-level and mixed, the number of topics may be increased. Or if there are several words appearing in many topics, they may be removed from the corpus. The user may also add or remove seed documents as necessary.

B. Discussion of the Human-in-the-Loop Methodology

The key idea behind the human-in-the-loop methodology is that as someone gains expertise in an area, his or her language changes to reflect it. More specifically, an expert in systems thinking will use language that reflects their expertise; thus the presence of the dimensions of systems thinking demonstrates strong systems thinking.

The methodology is fundamentally explorative. A certain document that exhibits strong systems thinking may not be tagged in this system thinking topic at all if the focus and domain of the document is very different than the rest of the corpus. This leads to a subjective and exploratory methodology opposed to a prescriptive and predictive tool.

1) Advantages and Limitations: This methodology has several advantages over the supervised method used by Whitehead et al. First, this process does not require 60% to 80% of the papers of interest to be manually graded; instead it only requires a few additional papers that the practitioner feels represent good systems thinking. This makes the methodology rapid and fast, while allowing for system thinking seeds to be changed quickly if the output is unsatisfactory.

Topic models also offer advantages over traditional methods of exploring textual information. While directly counting the frequency of the dimensions of systems thinking would provide some understanding of the potential for systems thinking in each document, the topic model is more general. The topic model utilizes the colocation of words; this allows the systems thinking topic to be corpus specific and to naturally find the domain-specific synonyms for the dimensions of systems thinking.

One direct limitation of this methodology emerges from the bag-of-words model of text. A document can use the correct words and have a large tagged proportion in the systems thinking topic, but not say anything meaningful or provide bad analysis. For this reason, we refer to the topic proportion as a proxy measure for the potential for systems thinking.

Additionally, this methodology is limited by the subjective nature of what defines good systems thinking and the subjective interpretation of topics within topic models, though the prior supervised method suffered this same limitation. On the other hand, the subjective nature of the human-in-the-loop iterative method balances potential biases introduced with the inherent capability of human intuition and expertise for handling abstract concepts more effectively than computers. The human-in-the-loop subjectivity does not detract from the numerical objectivity of the method.

2). Considerations and Insights: There are many simple considerations and insights that will benefit the users of the human-in-the-loop methodology.

Getting the systems thinking topic to emerge can be difficult in practice, and not all systems thinking topics are equally effective for analysis. If the system's thinking topic is too tightly clustered, no non-seed documents will have significant proportions in it. However, if it is made too general, the proportions will contain no valuable information. Large corpora tend to require more seed documents and more fine-tuning to acquire effective systems thinking topics; the increase in available data allows the topic model to find more potential patterns and clusters. The best way to evaluate the effectiveness of a systems thinking topic is inspection. A user can begin by inspecting the document topic proportions as they quickly provide insight into the question of the topic spread. Nonetheless, the ultimate validation of a systems thinking topic comes from inspection of the actual documents.

This methodology cannot escape the "garbage in, garbage out" conundrum of all data science. If a corpus is very disjoint, or very small, or not representative of the domain of interest the methodology may not work.

The user should be aware that not every paper *should* exhibit systems thinking. A report that is simply describing an event may not offer any analysis; it should not be surprising if it has a low proportion of systems thinking. Again, this does not make our methodology useless in understanding the system.

Iteration is often necessary to understand the corpus and to make the systems thinking topic emerge, but it is necessary from a validation standpoint as well. Topic models are stochastic methods that rely on random number seeds and approximations; values can change between iterations without changing any parameters or data. There is not a "true" set of topics and proportions that best describe the corpus; all topic models are wrong as they oversimplify language. Iteration serves to validate that the information coming from a topic model is reasonable and not a stochastic fluke.

All of these considerations and insights paint the picture of this methodology being an exploratory starting point; perfect for an analyst being introduced to a field with a minimal amount of time for reading or an analyst that is struggling to find the information they need.

C. Exploring a System with the Methodology

The goal of identifying strong systems thinking is valuable because the strong systems thinking of others can be useful in our own systems problems and systems design. A hallmark of strong systems thinking is the proper understanding of the system. Therefore, a natural extension of the human-in-the-loop methodology is to use the topic model for the general insight it provides of a corpus. A result is further understanding of the corpus and its content, in understanding of how documents are related, in understanding of central themes and connections of the system, and in research direction for the researcher. A user would follow the same process as the human-in-the-loop methodology for identifying documents with systems thinking, with options to change the seed documents and additional steps to interactively visualize the system after the naming of the topics.

Regarding the seeding, the user may seed documents that demonstrate or discuss strong systems thinking, use seed documents related to another topic or concept, or use no seeds at all. Each of these options will unveil different aspects about the system described by the corpus. Just as systems thinking seeds will lead to a systems thinking topic, which provides information about the systems thinking in each document, alternative seeds may lead to alternative topics providing similar information. The use of no seeds will describe the corpus as a whole, agnostic of any influence.

When the user reaches the step where he or she inspects and names the topics, he or she will naturally gain insight about the concepts within the corpus. This insight often leads to questions about how the topics are related, what ways the documents are interconnected, and how the ideas in the corpus are being used in practice. These questions can be answered through network graphs and through further iteration.

In creating a network graph of the topic model, each document is treated as a node, and the similarities between their thematic content is calculated using the topic proportions from the model (a document-to-document network). Several metrics have been proposed for this similarity measure, including the Kullback Leibler divergence, the symmetrized Jensen-Shannon (JS) divergence, Euclidian distance, the dot product, or cosine similarity [18]. Similarly, the user can use the topic proportions to make a network graph of the documents and their relationships to topics, treating both documents and topics as nodes (a document-to-topic network). Both of these network graphs tend to display useful and informative information, especially when visualized in a clustering visualization, such as those discussed in [19], [20], or [21]. One limitation of this visualization approach is that in inference most topic models never give true zero values for any of topic proportions. A possible result is an over-connected graph. This hurdle can be overcome by using a threshold function to keep only the most dominant edges.

Treating the topics as nodes, the user can make a similar network graph by finding the strength of their relationships through the documents they tag (a topic-to-topic network). Similar metrics as discussed for the document-to-document similarity can be used, though the vectors would come from the transpose of the document-topic matrix. As different topics may correspond to differing proportions of the corpus, this type of graph is typically dominated by the more structural and prevalent topics. This can be normalized and provide a more holistic view of the relationships between topics by keeping only the strongest N edges for each node (where N is a positive integer between one and the number of topics minus one).

V. Human-in-the-Loop Methodology Performance On Graded Corpus used in Prior work

We evaluated the effectiveness of the human-in-the-loop methodology at identifying documents with strong systems thinking by using the method on the manually graded corpus presented by Whitehead et al.

We added 5 systems thinking seeds (TABLE V in the appendix) to the 233 IEEE papers graded from 1 (bad systems thinking) to 10 (good systems thinking) and ran topic models with 10, 12, 15, 20, 25, and 30 topics for 2000 collapsed Gibbs sampling iterations using default stop words. These initial topic models did not produce a clear systems thinking topic as they were skewed by the structural words of the paper, so we repeated the models but removed additional words like “fig” “vol” “ieee” “model”, “university, and “results” as well as every mention of “IEEE Systems Journal”. TABLE VI in the appendix provides the full list of additional removed words. These models each produced an effective systems thinking topic (though an additional, less prevalent structural topic emerged too).

We choose to use the model with 15 topics for our comparison as its systems thinking topic seemed the most salient; TABLE II offers the top words and weights of the topic in table form and Fig. 3 visualizes the topic in a word cloud. While the dimensions of systems thinking as enumerated by Whitehead et al. do not overly dominate this topic, we believe the words it contains clearly hint at systems thinking and systems analysis. Additionally, we believe it contains other strong words that supplement the dimensions of systems thinking. For example, the words “factors” and “decision” may hint at the authors’ understanding of tradeoffs and traceability.

TABLE II

Top Words and Corresponding Weights for Emerging Systems Thinking Topic From Seeded Corpus

Word	Weight
Systems	3590
System	2513
Engineering	1101
Sos	955
Analysis	853
Management	643
Complexity	595
Architecture	529
Level	524
Requirements	515
Cost	496
Development	491
Factors	449
Decision	427
Thinking	372

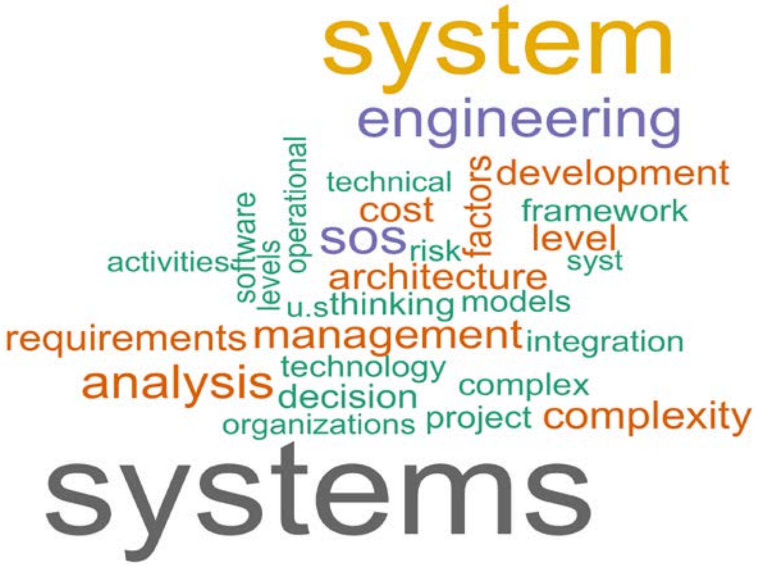


Fig. 3 Word cloud of the emerging systems thinking topic from the graded corpus seeded with documents that demonstrate and describe strong systems thinking

Fig. 4 visualizes the modeled graded corpus with seed documents in a network graph using the dot product as a measure of similarity between documents and the topic proportion as a measure of similarity between topics and documents. The network graph demonstrates the effectiveness of the seeding methodology to generate a systems thinking topic and to pull the documents with strong systems thinking (seeds and others) into the cluster of the visualization.

No structural topics were removed for the network graph. We only included edges with a similarity measure greater than 0.244 in the graph to focus on the true similarities. The golden nodes represent topics from the model and are labeled with their given name. The grey nodes represent regular documents, and the red nodes represent the seed documents that we felt demonstrated strong systems thinking. The names of the grey nodes are not displayed for readability, while the red nodes are only labeled with "TGT" for "target". The graph was visualized using the OpenOrd algorithm with Gephi [20], [22].

Fig. 5 shows a comparative boxplot of the system thinking topic proportions grouped by the documents grade. Papers with a grade of 9 or 10 appear to have significantly higher proportions in the systems thinking topic when compared to the other scores, though there are outliers for each grade group.

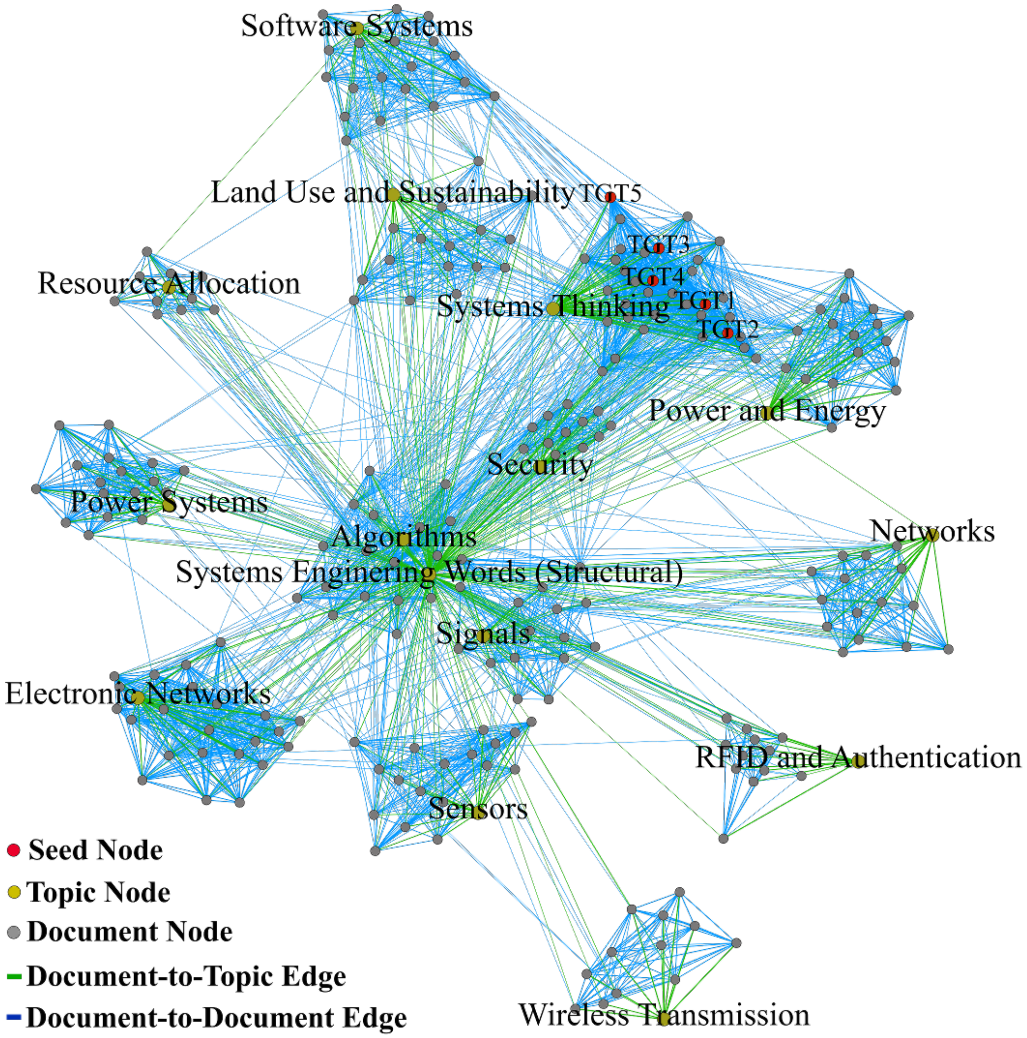


Fig. 4 Network graph visualizing IEEE graded corpus with seed documents. Notice the systems thinking cluster in the top-right of the graph and how all of the seed documents are pulled tight to it in clustering visualization.

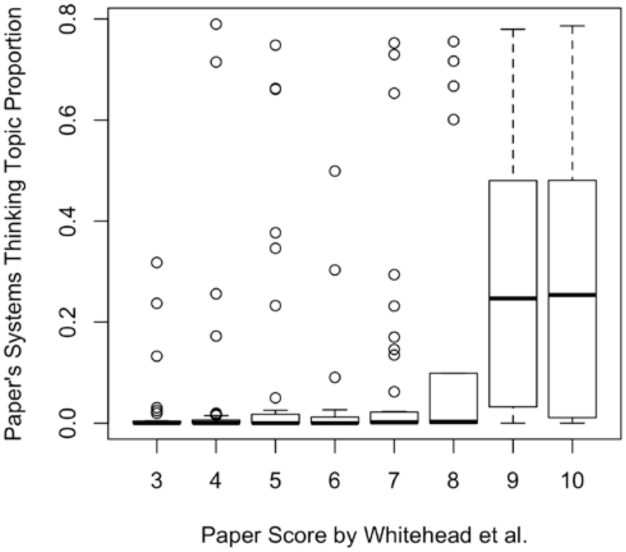


Fig. 5 Comparative boxplot showing distribution of emerging systems thinking topic within graded corpus. Papers scored with either a 9 or 10 have much higher proportions in the systems thinking topic than the papers with lower grades, though there are many outliers for the other groups with high proportions in the topic.

To test this statistically, we performed a Tukey test on the papers grouped by their manual score. The mean topic proportion in the systems thinking topic for the groups of papers graded 9 and 10 was statistically higher than the groups of papers graded 3, 4, 5, 6, or 7 at the 0.05 level. There was no statistically significant difference at the 0.05 level between the means of the groups graded 3, 4, 5, 6, 7, or 8 or between the groups graded 8, 9, or 10. TABLE III details the p values for this test.

TABLE III
P Values Of Tukey Paired Mean Test Between Papers Grouped By Manual Grade

		Paper Grade							
		3	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Paper Grade	**3**		1.00	0.88	1.00	0.91	0.24	**<0.01**	**<0.01**
	4	1.00		0.92	1.00	0.95	0.25	**<0.01**	**<0.01**
	5	0.88	0.92		0.95	1.00	0.88	**0.02**	**0.02**
	6	1.00	1.00	0.95		0.97	0.33	**<0.01**	**<0.01**
	7	0.91	0.95	1.00	0.97		0.82	**0.01**	**0.01**
	8	0.24	0.25	0.88	0.33	0.82		0.44	0.57
	9	**<0.01**	**<0.01**	**0.02**	**<0.01**	**0.01**	0.44		1.00
	10	**<0.01**	**<0.01**	**0.02**	**<0.01**	**0.01**	0.57	1.00	

Bold signifies that means of groups are statistically significant at the 0.05 level.

The Tukey test and the comparative boxplot suggest that a practitioner would find strong systems thinking much more quickly using this methodology than by manual exploration of a corpus; the strongest system thinking documents have higher topic proportions in the emerging systems thinking topic. This validates the usefulness of the methodology for identifying top documents that exhibit strong systems thinking, but it does not support the use of the methodology as a prescriptive technique. It cannot be used prescriptively since the topics must be evaluated manually and there is no natural way to set a threshold for classification as the topic proportion distribution is extremely dependent on the corpus content as well as the model parameters. The value in the methodology is the insight a practitioner would gain from exploring the results. Finally, this validation does not overcome the inherent subjectivity in what defines strong systems thinking, but we believe that this is why the iterative nature and the human-in-the-loop are essential. People can handle subjectivity better than computers and algorithms.

VI. Army Human Dimension Case Study

The US Army Training and Doctrine Analysis Center (TRAC) in Monterey, California provided a dataset of 180 documents related to human dimension projects, problems, and studies within the Army for a case study of the human-in-the-loop methodology. The Army was specifically interested in identifying where systems thinking was occurring within this domain and summarizing the system.

We performed several iterations with the human-in-the-loop methodology using 20, 25, 30, and 35 topics and seeding the corpus with the four documents, listed in the appendix in TABLE V that we felt described strong systems thinking. Before topic modeling, we also removed the default stop words provided by Mallet. We felt that the emerging systems thinking topic was strongest in the model with 25 topics, so it was used for all following analysis.

We then named each of the 25 topics by manually evaluating each topic's word weights and the corresponding word cloud. TABLE IV and Fig. 6 provide an example of the topic that was named "Physical and Psychological Resilience" as the word describe physical and psychological factors with a strong weight on the word "resilience". The table lists the top 15 words in the topic with their corresponding weights while the figure shows the word cloud visualization of the topic.

TABLE IV

TOP WORDS AND CORRESPONDING WEIGHTS FOR TOPIC NAMED "PHYSICAL AND PSYCHOLOGICAL RESILIENCE"

Word	Weight
Resilience	520
Stress	263
Force	245
Health	228
Air	203
Fitness	198
Social	163
Factors	157
Physical	129
Psychological	126
Scale	115
Stressors	101
Resources	97
Measures	94
Well-being	92
Research	89
Mental	88

The structural word "vol" was excluded from the topic's table as it was heavily prevalent in all topics.

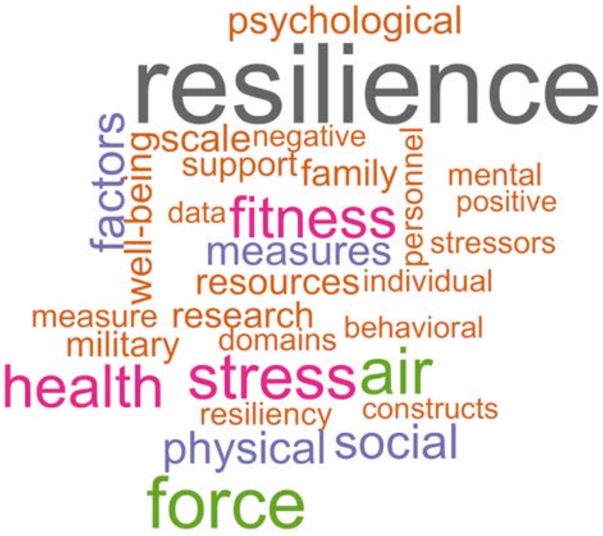


Fig. 6 Word cloud of the "Physical and Psychological Resilience" Topic. Topics can be visualized in word clouds to facilitate naming. The structural words "vol", "e.g.", "literature", and "journal" were removed from the word cloud as they were heavily prevalent in all topics.

We created a network graph of the corpus, which is visualized in Fig. 7, using the dot product as measure of similarity between documents and the topic proportion as a measure of similarity between topics and documents. In order to focus on only the dominant and informative relationships, we removed the structure topic that was dominated by words like "research", "university", "table", and "figure" from the network, and we only included edges with a similarity measure greater than 0.05. The node colors, edge colors, and visualization algorithm are the same as used for the prior network graph.

This network visualization provides a summary of the domain, and we offer the interpretation that everything the Army does in the human dimension is driven by the goal of making men and women leaders. This is supported by the centrality of the "Army Leaders" topic; almost all other clusters have connections to this topic. This leads to the interpretation that the other emerging clusters represent areas where the Army is trying to make men and women leaders or methods they are using to research and do this. Furthermore, it demonstrates the effectiveness of the seeding methodology to create a custom topic.

This use of the human-in-the-loop methodology for summarization highlights several valuable features. First, the methodology does not require a subject matter expert to generate an overview, though a subject matter expert could potentially find more in-depth analysis. We are not experts on the military domain in any capacity, and our analysis was done without the help of MAJ Connors. MAJ Connors agreed that our interpretation was extremely accurate and insightful. Second, the topic model is exhaustive and free from human bias. While a soldier offering an overview of human dimension research in the Army may have some bias towards certain themes or simply forget others, the topic model will not. Finally, the summary it provides is effective and allows the user to quickly ask more questions to delve deeper into the material; documents can be read based on their position in the network graph or based on their proportion in topics of interest.

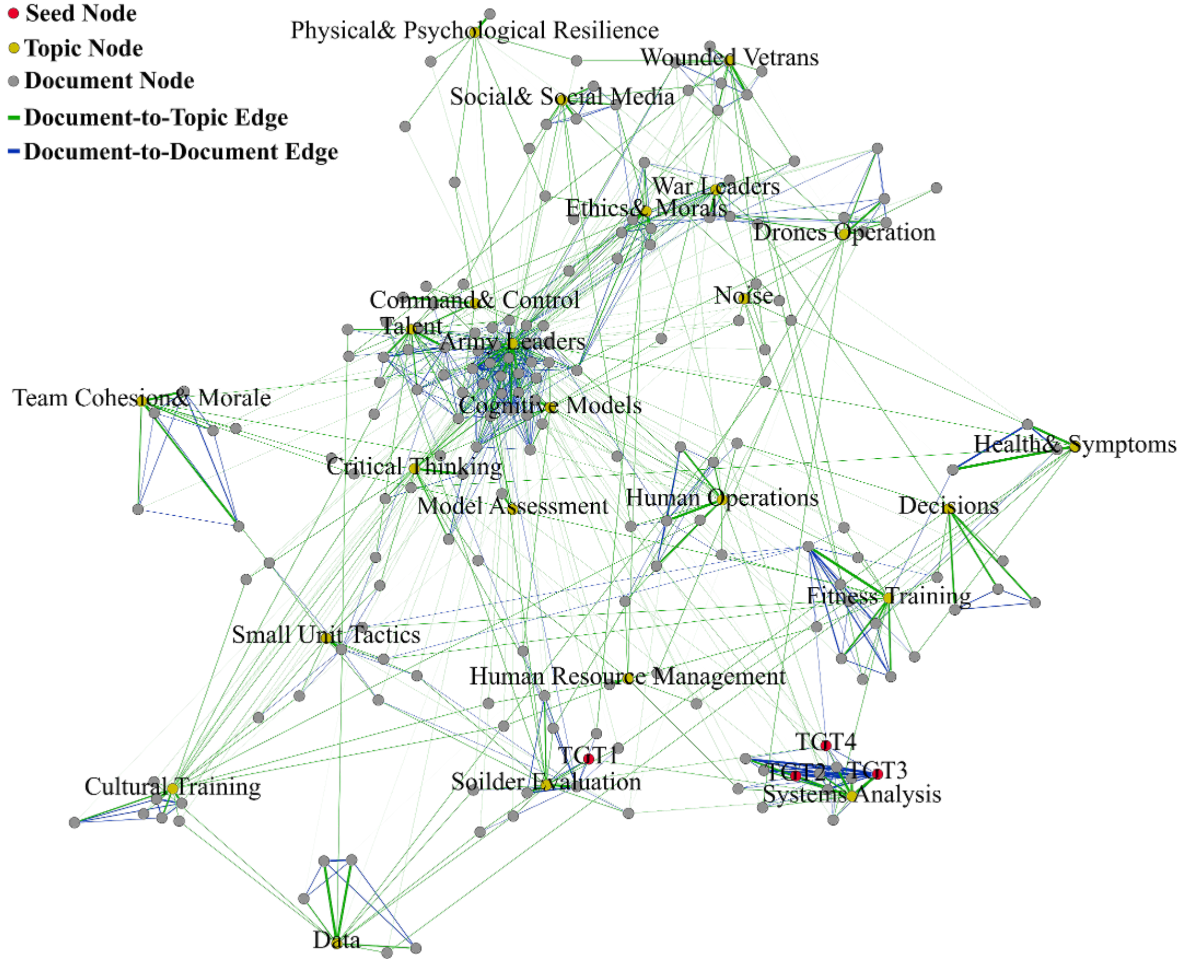


Fig. 7 Network graph showing army human dimension corpus. Note the systems thinking and systems analysis cluster in the bottom left corner with the majority of the seed documents gathered to it

A systems thinking topic emerged from the methodology, bringing in three of the four seed documents and around seven other documents in the cluster visualization. While this cluster demonstrates the potential of the methodology to generate a systems thinking topic, it does not validate that the measurement of systems thinking is valuable.

Despite the subjectivity of strong systems thinking, a group of four of us (two from the Department of Systems and Information Engineering at the University of Virginia and two from the Army Training and Doctrine Analysis Center in Monterey, California) read the five papers with the largest proportion in the systems thinking topic and the five papers with the smallest proportion in the topic. We collectively agreed that the top five documents demonstrated good systems thinking: they were appropriately scoped, identified important stakeholders, discussed decisions in terms of tradeoffs, and provided traceable arguments. Additionally, we agreed that the bottom five papers did not demonstrate strong systems thinking. About half of these bottom papers did not have potential for systems thinking; they were simply recounting an event or presenting information. However, we felt that the few papers in the bottom five that provided analysis lacked strong systems thinking. Once again, this judgment is very subjective, but does provide some support for the human-in-the-loop methodology for identifying systems thinking.

Most of the papers in the top five and bottom five were heavily related to command and control, hinting that there is no clear area where systems thinking is especially strong or weak within the human dimension in the Army. A possible interpretation of this is that the Army needs system thinking the most within command and control.

Overall, this case study demonstrates that human-in-the-loop methodology can be extremely useful for understanding a system, and can be effective in alternative domains with alternative, domain-specific vocabularies.

VII. Future Work and Conclusion

There is opportunity to continue building on the human-in-the-loop methodology; potential future work includes performing a true designed experiment with analysts to validate and quantify the effects of the methodology on workflow. This could be done by giving a collection of analysts a corpus which they have not seen and asking them to find specific information or summarize it effectively; half could be given the tool with training, and half could use whatever alternative methods they prefer. Additionally, more research could be done on modifying the actual topic model algorithm to increase its effectiveness in identifying systems thinking.

Overall, the human-in-the-loop methodology shows tremendous ability to leverage topic models to identify systems thinking and to assist in the exploration and understanding of a system. This can be valuable to system practitioners everywhere as it will facilitate their systems analysis and allow them to more easily learn from those who came before them. There is still tremendous subjectivity in the definition of strong systems thinking and the interpretation of the topics, but this is why the human-in-the-loop methodology is so effective. Computers can handle the processing of the data while human intuition handles the subjective nature and tailors the results to the problems the practitioner cares about.

Appendix

Documents Used as Seeds in Graded Corpus Exploration and Human Dimension within the Army Case Study.

TABLE V
TABLE OF SEED DOCUMENTS

Document	Used in		Description or Citation
	Graded Corpus	Human Dimension Corpus	
Dimensions of Systems Thinking	TGT1	TGT1	List of the dimensions of systems thinking from [1]
Chapter 10 – How To Do Systems Analysis	TGT2	TGT2	[5]
System’s Thinking about Systems Thinking: A Proposal for a Common Language	TGT3	TGT3	[1]
System’s Thinking Word List		TGT4	Custom list of potential synonyms for the dimensions of systems thinking
Perspectives of the Systems Approach	TGT4		[23]
Hexagons for Systems Thinking	TGT5		[24]

Additional Stop Words and Stop Phrases Removed From IEEE Graded Corpus From Whitehead et. al.

TABLE VI
ADDITIONAL WORDS REMOVED FROM GRADED CORPUS (CASE INSENSITIVE)”

Word		
Approach	Paper	January
Based	Performance	February
Case	Problem	March
Data	Proc	April
Degree	Process	May
Due	Research	June
Fig	Results	July
Function	Section	August
IEEE	Senior Member IEEE	September
IEEE Systems Journal	Set	October
Information	Shown	November
Introduction	Table	December
Journal	Time	
Life Fellow IEEE	University	
Member IEEE	Vol	
Model	Work	
Number		
Order		

II. Works Cited

[1] N Peter Whitehead, William T Scherer, and Michael C Smith, "Systems Thinking About Systems Thinking A Proposal for a Common Language," *IEEE Systems Journal*, vol. 9, no. 4, pp. 1-12, December 2015.

[2] N Peter Whitehead, William T Scherer, and Michael C Smith, "Use of Natural Language Processing to Discover Evidence of Systems Thinking," *IEEE Systems Journal*, vol. PP, p. 2015, May 2015.

[3] Russell L Ackoff, "Towards a System of Systems Concepts," *Management Science*, vol. 17, no. 11, pp. 661-671, 1971.

[4] C West Churchman, *The Systems Approach*. New York, USA: Delacorte Press, 1968.

[5] John E Gibson, William T Scherer, and William F Gibson, *How To Do Systems Analysis*. Hoboken, New Jersey, USA: John Wiley & Sons, Inc., 2007.

[6] David Blei, "Probabilistic Topic Models," *Communications of the ACM*, vol. 55, no. 4, April 2012.

[7] David M Blei, Andrew Y. Ng, and Michael I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, pp. 993-1022, January 2003.

[8] Stuart Geman and Donald Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence* , vol. PAMI-6, no. 6, pp. 721-741, November 1984.

[9] Thomas P Minka, "Expectation Propagation for Approximate Bayesian Inference," *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*, pp. 362-269, August 2001.

[10] Hanna M Wallach, David Mimno, and Andrew McCallum, "Rethinking LDA: Why Priors Matter," *Advances in Neural Information Processing Systems 22*, pp. 1973-1981, 2009.

[11] Michael A Livermore, Allen B Riddel, and Daniel Rockmore, "Graphical Models, Exponential Families, and Variational Inference," *Available at SSRN*, 2016.

[12] David M Blei and John D Lafferty, "A Correlated Topic Model of Science," *The Annals of Applied Statistics*, vol. 1, no. 1, pp. 17-35, June 2007.

[13] Xing Wei and W. Bruce Croft, "LDA-Based Document Models For Ad-hoc Retrieval ," *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 178-185, 2006.

[14] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D Manning, "Labeled LDA: A Supervised Topic Model for Credit Attribution in Multi-Labeled Corpora," *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pp. 248-256, 2009.

[15] Andrew Kachites McCallum. (2002) MALLET: A Machine Learning for Language Toolkit. [Online]. http://mallet.cs.umass.edu

[16] Yee Whye Teh, Michael L Jordan, Matthew J Beal, and David M Blei, "Hierarchical Dirichlet Processes," *Journal of the American Statatistical Association*, vol. 101, no. 476, pp. 1566–1581, 2006.

[17] Thomas L Griffiths and Mark Steyvers, "Finding Scientific Topics," *Proceedings of the National Academy of the Sciences*, vol. 101, no. 1, pp. 5228–5235, April 2004.

[18] Mark Steyvers and Tom Griffiths, "Probabilistic Topic Models," in *Latent Semantic Analysis: A Road to Meaning*. Mahwah, New Jersey, USA: Laurence Erlbaum, 2007.

[19] Thomas M. J. Fruchterman and Edward M. Reingold, "Graph Drawing by Force-Directed Placement," *Journal of Software: Pracitce and Experience*, vol. 21, no. 11, pp. 1129-1164, November 1991.

[20] Shawn Martin, W. Michael Brown, Richard Klavans, and Kevin Boyack, "OpenOrd: An Open-Source Toolbox for Large Graph Layout," *Proceedings of SPIE - The International Society for Optical Engineering*, January 2011.

[21] Yifan Hu, "Efficient, High-Quality Force-Directed Graph Drawing," *The Mathematica Journal*, vol. 10, no. 1, pp. 37-71, 2006.

[22] M Bastian, S Heymann, and M Jacomy, "Gephi: An Open Source Software for Exploring and Manipulating Networks," *International AAAI Conference on Weblogs and Social Media*, 2009.

[23] C West Churchman, "Perspectives of the Systems Approach," *Interfaces*, vol. 4, no. 4, pp. 6-11, August 1974.

[24] Anthony M Hodgson, "Hexagons for Systems Thinking," *European Journal of Operational Research*, vol. 59, no. 1, pp. 220-230, May 1992.

Appendix C Shiny Topic Models User Manual

Shiny Topic Models User Manual
Ryan Boyer
August 2016

Table of Contents:

List of Graphs and Figures

1. Overview

Shiny Topic Models is an R Shiny Application created to streamline the process of creating and exploring topic models. The software allows users to run and analyze topic models or to import already run topic models and analyze them interactively. Key features include the exploration of topics, the exploration of data, and the making of network graph files to be viewed in an external network graph viewer.

Functionally, Shiny Topic Models utilizes Mallet, and open source topic modeling java package. This allows the topic model run by Shiny Topic Models to be fast, efficient, and light.

2. Set-Up and Installation

2.1. Necessary Software

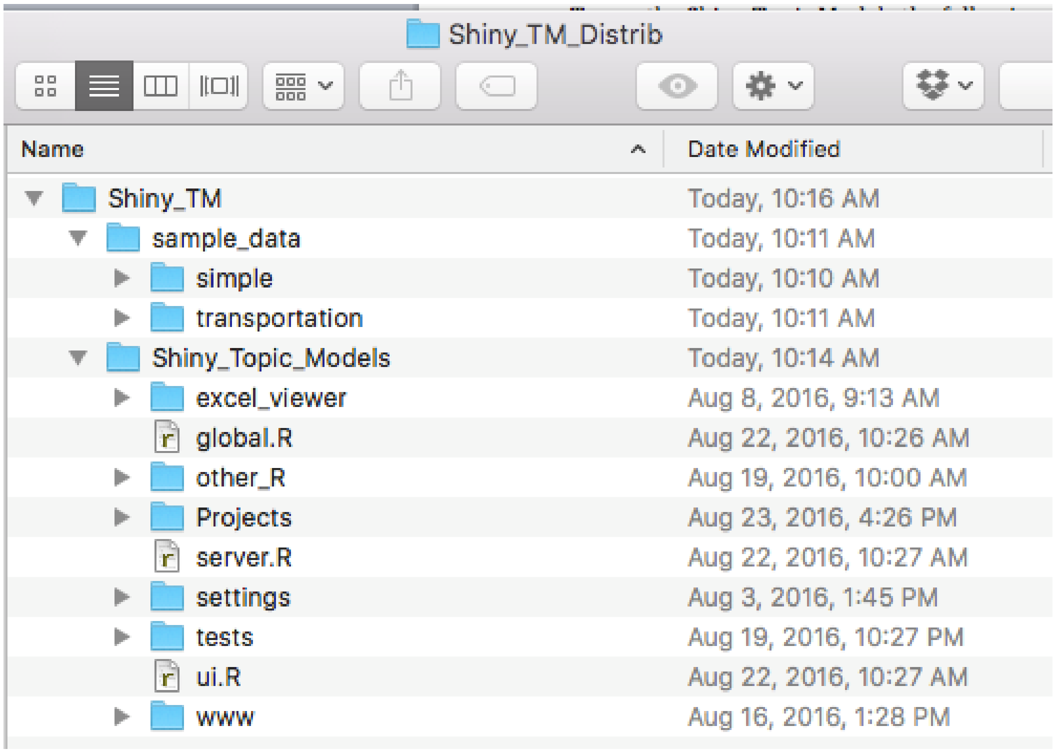
To use the Shiny Topic Models the following software must be installed:

- Java SDK
  - Download Link: http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
  - This is NOT the JRE, you need the JDK (Developer Kit, not Runtime Environment)
  - To check if you have installed already:
    - Mac "`javac -version`" in terminal
    - Windows see http://stackoverflow.com/questions/4681090/how-do-i-find-where-jdk-is-installed-on-my-windows-machine
- Mallet
  - Download Link: http://mallet.cs.umass.edu/download.php
  - Either the tar.gz or .zip file is fine (as long as you unpack it)
  - Place uncompressed files and folders in a safe location on your computer
  - DO NOT CHANGE the file structure of the files within the uncompressed folder
- R
  - Download Link: https://cran.r-project.org
- R Studio (Recommended IDE for R)
  - Download Link https://www.rstudio.com/products/rstudio/download3/
  - Free Studio Desktop version is fine.
- Gephi (For viewing network graph files)
  - Download Link: https://gephi.org
- XQuartz (Mac only, necessary for viewing save file dialog)
  - Download Link: https://www.xquartz.org

2.2. Installing Shiny Topic Models

a) Decompress the Shiny_TM file and move the resulting folder and all of its contents to the place you would like to store it

   a. Mallet **CANNOT** have spaces in the file path names, so move it to a location that is suitable.



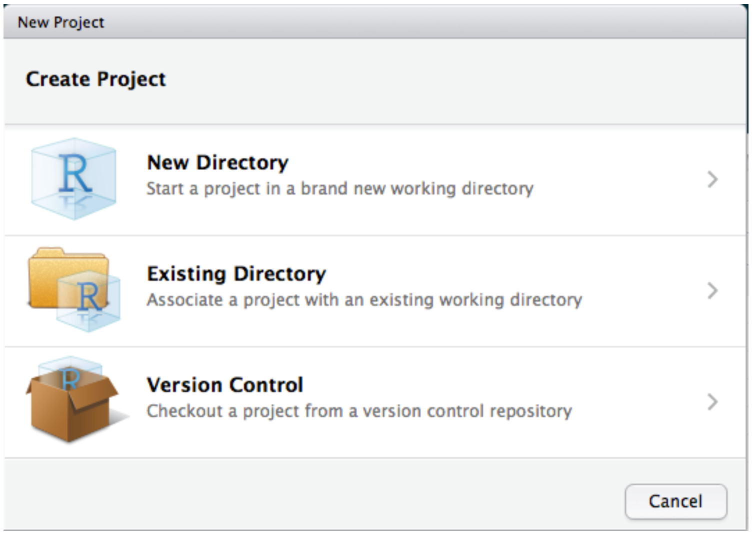
Shiny_TM_Distrib

Name	Date Modified
Shiny_TM	Today, 10:16 AM
sample_data	Today, 10:11 AM
simple	Today, 10:10 AM
transportation	Today, 10:11 AM
Shiny_Topic_Models	Today, 10:14 AM
excel_viewer	Aug 8, 2016, 9:13 AM
global.R	Aug 22, 2016, 10:26 AM
other_R	Aug 19, 2016, 10:00 AM
Projects	Aug 23, 2016, 4:26 PM
server.R	Aug 22, 2016, 10:27 AM
settings	Aug 3, 2016, 1:45 PM
tests	Aug 19, 2016, 10:27 PM
ui.R	Aug 22, 2016, 10:27 AM
www	Aug 16, 2016, 1:28 PM

Figure 1: Overview of files in Shiny_TM compressed file

b) Open RStudio

c) Select File → New Project. The new project window will open. (If this is the first time opening RStudio, the new project window should open automatically.)



New Project

Create Project

New Directory
Start a project in a brand new working directory

Existing Directory
Associate a project with an existing working directory

Version Control
Checkout a project from a version control repository

Figure 2 New Project Window in RStudio

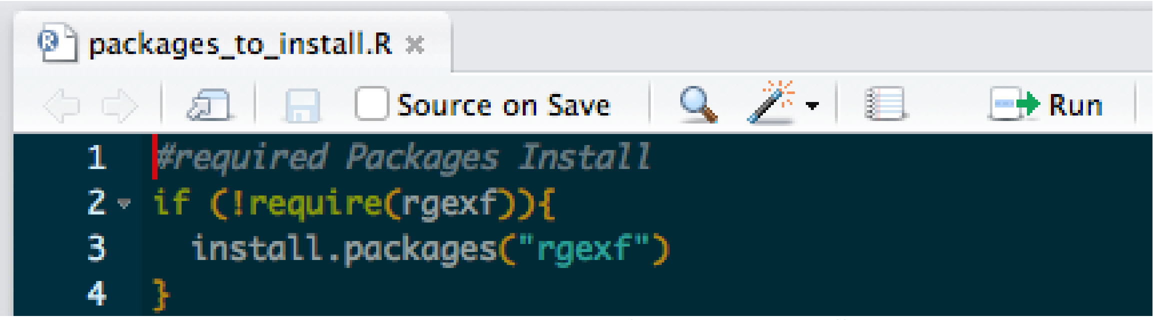
d) Select "Existing Directory"

e) Select "Browse". Navigate to the directory where you uncompressed the Shiny_TM file. Select the "Shiny_Topic_Models" folder as seen in Figure 1.

f) Select "Create Project" RStudio will initialize a new project and open it.

g) In the files panel in RStudio (Default bottom right block, left most tab), double click "packages_to_install.R" to open the file.

h) Select all text in the file within the editor and then click run. This will download and install the necessary R packages. It will not download those already installed.

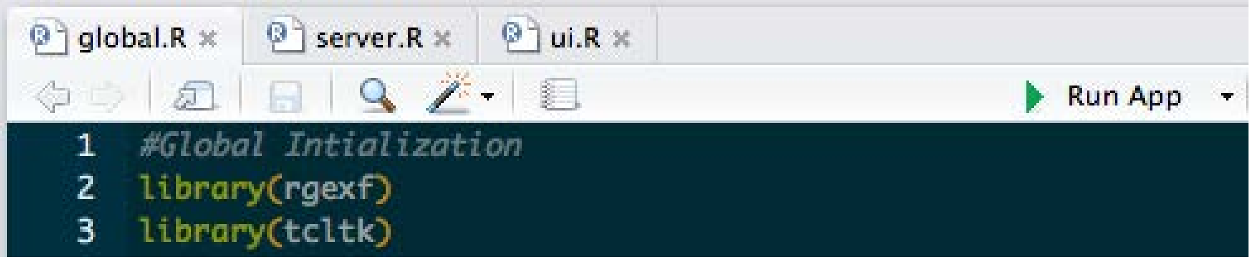


```
#required Packages Install
if (!require(rgexf)){
  install.packages("rgexf")
}
```

Figure 3 "packages_to_install.R" Note the Run button in the top left. Color scheme may differ between RStudio versions.

i) In the files panel in RStudio (Default bottom right block, left most tab), double click on "global.R", "server.R", and "ui.R" to open these files.

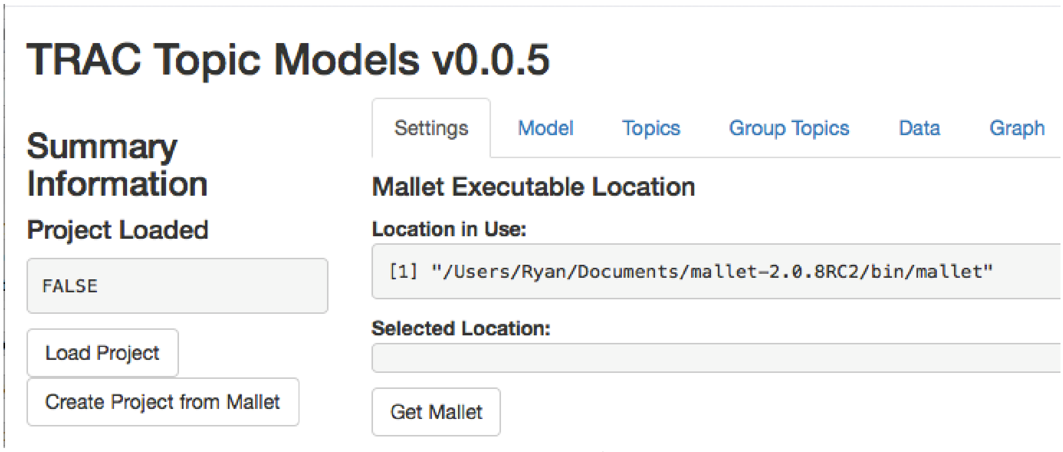
j) In the editor panel, select any of the three files just opened and click "Run App"



```
#Global Intialization
library(rgexf)
library(tcltk)
```

Figure 4 "Run App" Button

k) R will load the necessary packages and the settings file then open the App. This will either be within RStudio or a web browser depending on your operating system and preferences.



TRAC Topic Models v0.0.5

Summary Information

Project Loaded: FALSE

Mallet Executable Location

Location in Use: [1] "/Users/Ryan/Documents/mallet-2.0.8RC2/bin/mallet"

Selected Location:

Figure 5 Shiny Topic Models Default Screen

3. Using The Software

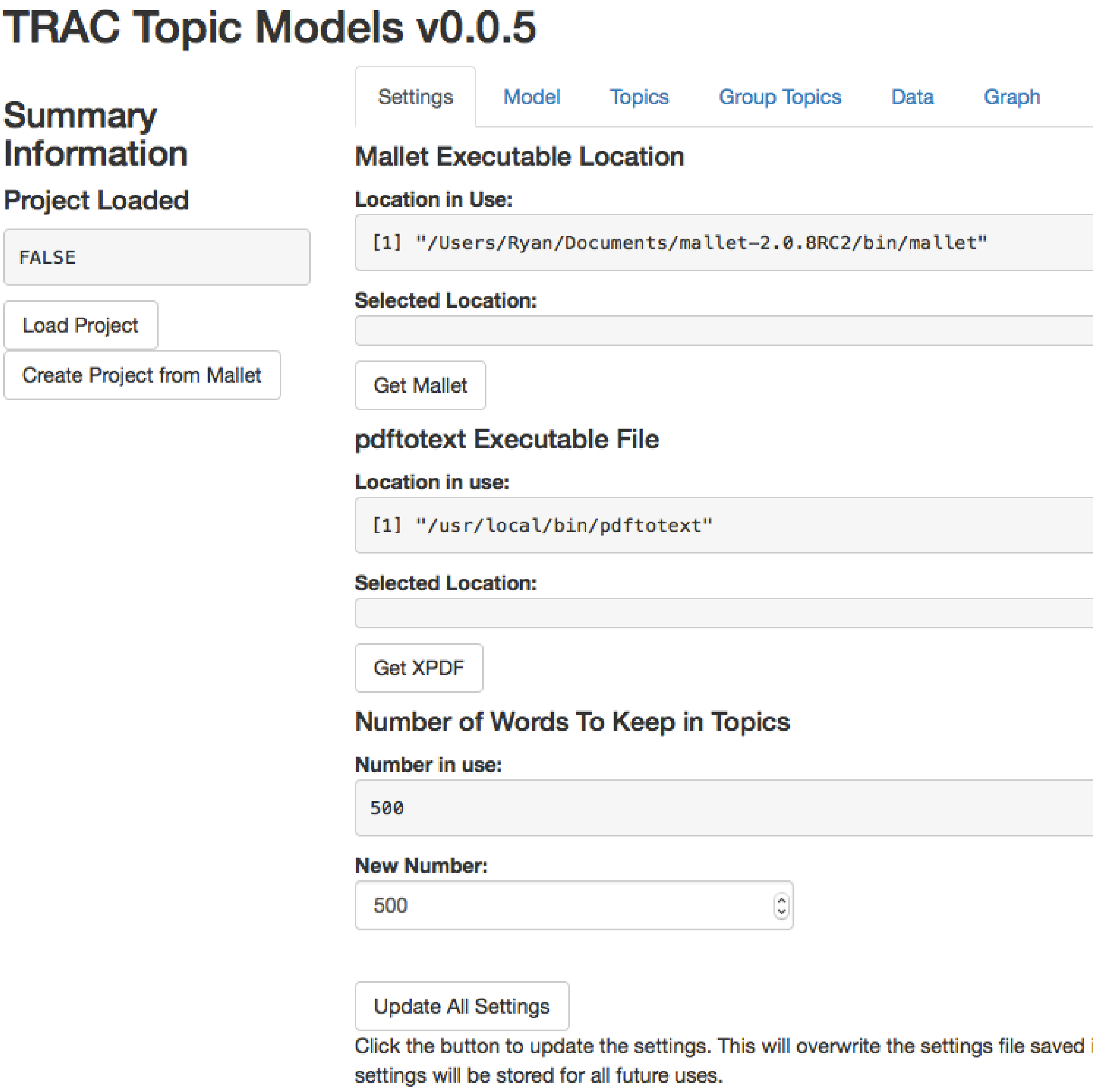
This section of the manual walks through the use of each tab of the Shiny Topic Models Application.

There are three major concerns that need to be addressed before continuing:

- Mallet cannot handle file paths with spaces. Any directory or extra stop words file passed to mallet cannot have a space anywhere in the file path. This is an inherent problem in Mallet (not shiny), and cannot be easily fixed without rebuilding the mallet infrastructure. Current work around is to store your files somewhere without spaces.
    - E.g. “/Users/LastName FirstName/files/mytextfiles/” would fail as there is a space between “lastname” and “firstname”
    - “/Users/LastName_FirstName/Files/mytextfiles/” would work as there is an underscore instead of a space
- **On Windows** the TCLTK **windows for getting save paths popup behind the focus windows of R Shiny and R**. Move windows around and look for a new Windows Explore window if you clicked a button to save a file or select a file or folder.
- **To open a project after closing Shiny Topic Models, you must select “Save Project” or “Save Project As” on the right sidebar before closing.** The many save buttons within the tabs save settings to the server, but DO NOT save it to the project file. The additional save step is required to retain these settings for future uses; this is where it is actually written to the computer. (This is intentional. For large topic models saving the file to the computer can take a long time.)

3.1. Settings

The settings tab allows the user to set and store the application settings. These are saved and loaded each time the application is run.



TRAC Topic Models v0.0.5

Summary Information

Project Loaded

```
FALSE
```

Mallet Executable Location

Location in Use:

```
[1] "/Users/Ryan/Documents/mallet-2.0.8RC2/bin/mallet"
```

Selected Location:

pdftotext Executable File

Location in use:

```
[1] "/usr/local/bin/pdftotext"
```

Selected Location:

Number of Words To Keep in Topics

Number in use:

```
500
```

New Number: 500

Click the button to update the settings. This will overwrite the settings file saved i
settings will be stored for all future uses.

Figure 6 Settings Tab

There are three settings. All of them must be in the correct format for them not to be overwritten on application load. **CURRENTLY THE “pdftotext” FUNCTIONALITY IS NOT IMPLEMENTED. SET THIS TO THE MALLET PATH ALSO FOR THE APPLICATION TO WORK.** You must click the “Update all Settings” button to save the settings.

a) **Mallet Executable Location**
This is the location of the mallet binary file from the mallet installation (from http://mallet.cs.umass.edu/download.php). This should be a file in the location <~/mallet-VERSIONNUMBER/bin/mallet>. Note that this is NOT the “mallet.bat” file.

To set the mallet path, click the “Get Mallet” button and then select the file.

b) Pdftotext Executable File
THIS FEATURE IS NOT IN USE, but is kept for future development.

Set this to the same file path as you did in step a; click the “Get XPDF” button and then select the file.

c) Number of Words to Keep in Topics
This is the number of words per topic that the topic model will keep for each topic. For instance, a collection of 300 documents may have 1,000,000 unique words. Each topic will have a value for each of those words. Setting this setting to 500 will only keep the top 500 words and their values for each topic, significantly reducing the size of the save file.

500 is the recommended setting.

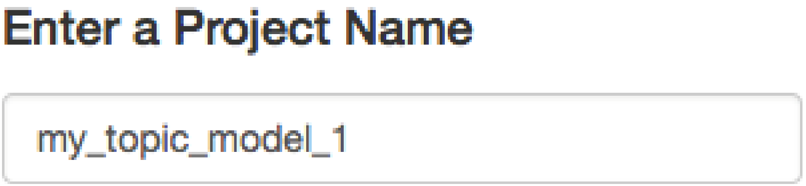
Click on the arrows or type a number in the box to change the value.

d) Click “Update All Settings” to save the settings to file.

3.2. Model

The model tab will run a topic model on selected text files and generate a file format appropriate for the Shiny Topic Models application. The tab allows the user to select a project name, data, model settings, and stop words for the algorithm. Each of these settings is explained below. Once all of the settings are chosen, the user selects the blue “RUN” button to run the algorithm.

a) **Project Name**
Type a descriptive name for the project in the text box. This will be used to name the project folder when stored on your system.

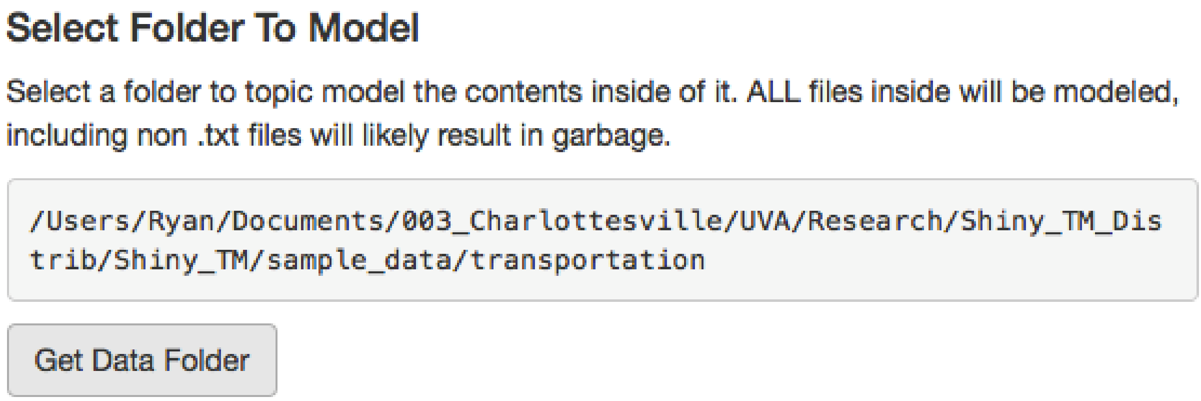


Enter a Project Name

my_topic_model_1

Figure 7 Project Name

b) **Folder to Model**
Click the “Get Data Folder” button to select a folder full of .txt files to topic model. All files in the folder must be “.txt” files or else the algorithm will return garbage. However, these files can be hosted in subfolders. See the sample data included for examples.



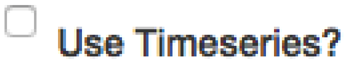
Select Folder To Model

Select a folder to topic model the contents inside of it. ALL files inside will be modeled, including non .txt files will likely result in garbage.

```
/Users/Ryan/Documents/003_Charlottesville/UVA/Research/Shiny_TM_Distrib/Shiny_TM/sample_data/transportation
```

Figure 8 Select Folder to Model

c) **Use Time Series**
Check the "Use Time series?" box if the subfolders of all text files are numeric, representing a time period that the files originated from. See the "transportation" sample data for an example. This will tag all text files with the appropriate year.



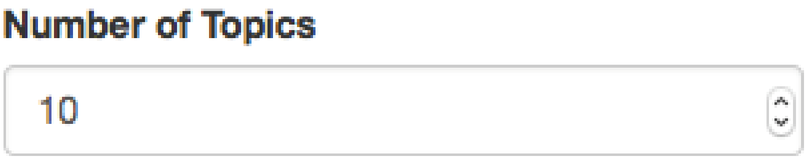
□ Use Timeseries?

Figure 9 Use Time Series

Currently this functionality is only visible when exploring the data, but future updates could result in dynamic time series graphs.

d) **Algorithm Settings**

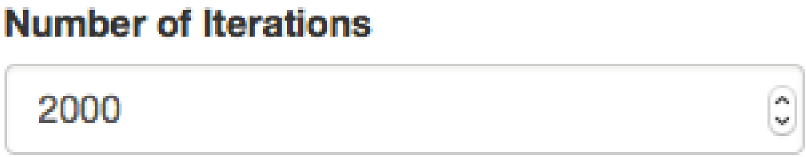
a. **Number of Topics**
Select the number of topics to find within your data. Use the selector by typing a number or using the arrow keys. You must select at least 2 topics. Typically, multiple iterations of topic models are necessary to get the best value for this setting.



Number of Topics: 10

Figure 10 Number of Topics

b. **Number of Iterations**
Select the number of iterations to run the topic model. Use the selector by typing a number or using the arrow keys. There exists a point where more iterations do not improve the algorithm's effectiveness or accuracy, though it is difficult to identify. 2000 is a safe selection. Obviously, more iterations makes it take longer.

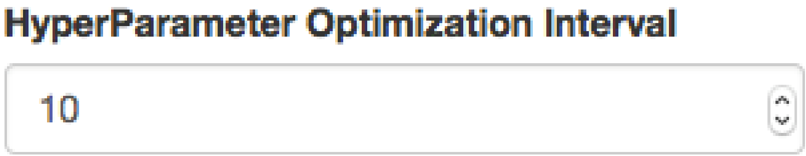


Number of Iterations

2000

Figure 11 Number of Iterations

c. **Hyper Parameter Optimization Interval**
   Select the interval at which to optimize the hyper parameter. Every X iterations of the regular LDA algorithm, mallet will attempt to optimize the model parameters. Use the selector by typing a number or using the arrow keys. 10 is the recommended selection, according to the Mallet creators.

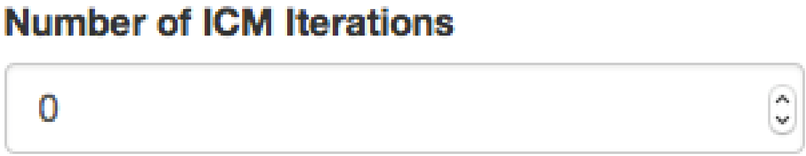


HyperParameter Optimization Interval

10

Figure 12 Hyper Parameter Optimization Interval

d. **Number of ICM Iterations**
   Select the number of Iterated Conditional Modes to run after completion of the topic model. This will further smooth the topics of the topic model, but generally is unnecessary. 0 is recommended selection.



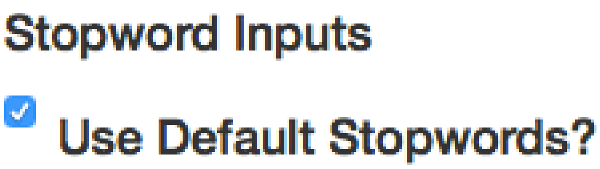
Number of ICM Iterations

0

Figure 13 Number of ICM Iterations

e) **Stop words**

a. **Default Stop Words**
   Check the box to use the default stop words. These common words are visible in the text file “~/mallet-<VERSIONNUMBER>/stoplists/en.txt” if you want to see them.



Stopword Inputs

☑ Use Default Stopwords?

Figure 14 Use Default Stop Words

b. **Extra Stop words**

Click the “Get txt file with Extra Stop words” button to select an additional file of stop words to use. Note that mallet casts everything into lower case, but does not automatically do this with additional stop words lists. Make all words in this additional list lowercase for it to be implemented correctly.

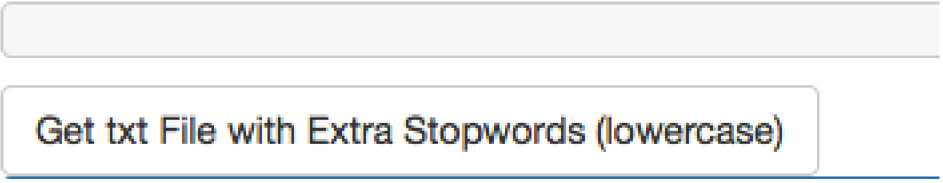


Figure 15 Additional Stop words Button

Do not select a file if you do not want any additional stop words.

f) **RUN**

Once all of the above settings are as desired, click the “RUN” button to run the topic model.

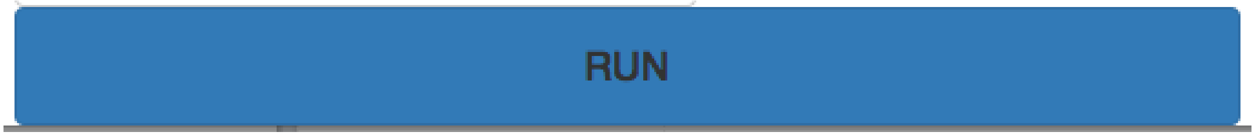


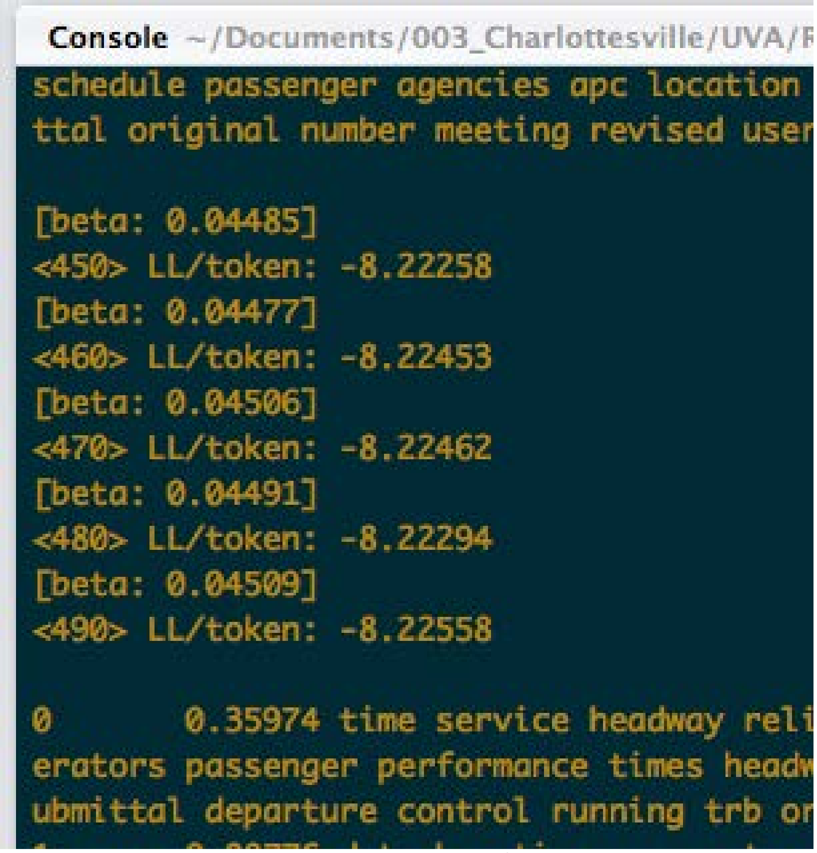
Figure 16 Run Button

Once clicked the following happens:

a. A project folder is created within the “Projects” folder of the Shiny Topic Models Package.

b. Mallet collects the txt files in the folder and imports them into a form that it can understand.

c. Mallet runs an LDA topic model (as dictated by the settings above) using a collapsed Gibbs sampler.
The progress of the algorithm is visible in the console of R Studio.



```
schedule passenger agencies apc location
ttal original number meeting revised user

[beta: 0.04485]
<450> LL/token: -8.22258
[beta: 0.04477]
<460> LL/token: -8.22453
[beta: 0.04506]
<470> LL/token: -8.22462
[beta: 0.04491]
<480> LL/token: -8.22294
[beta: 0.04509]
<490> LL/token: -8.22558

0       0.35974 time service headway reli
erators passenger performance times headw
ubmittal departure control running trb or
```

Figure 17 Algorithm Process in R Studio Console. The <###> identifies which iteration the algorithm has finished.

Unfortunately, it cannot be sent to the Shiny Topic Models Window.

d. The output of the mallet files are interpreted by R and imported into Shiny Topic Models; the large files are deleted.

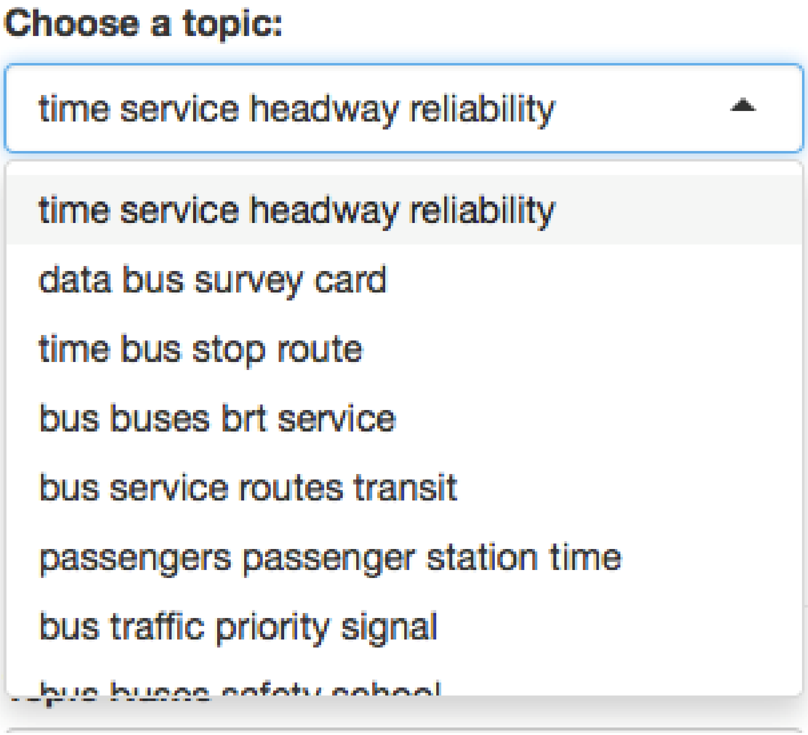
e. Shiny Topic Models saves the initial version of the project to the disk. This will be "~/Shiny_Topic_Models/Projects/<project_name>/data/project_file.JSON"

f. Shiny Topic Models creates a macro-enabled Excel file called "DocTopicsTemplate.xlsm" which can be used to view the topic model within excel. This is stored initially in the folder "~/Shiny_Topic_Models/Projects/<project_name>/excel/" and requires the "DocTopics.txt" and "TopicKey.txt" files to work correctly.

3.3. Explore Topics

The topics tab allows the user to explore topics in the model and name them as desired.

Select a topic using the drop down menu. Initially the topics are named using the top four words in the topics.

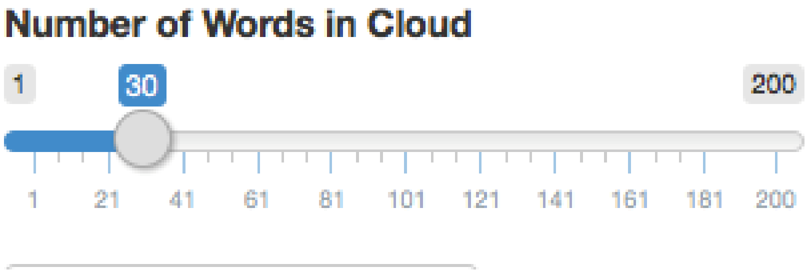

Choose a topic:

time service headway reliability

time service headway reliability
data bus survey card
time bus stop route
bus buses brt service
bus service routes transit
passengers passenger station time
bus traffic priority signal
bus buses safety school

Figure 18 Choose a Topic to View

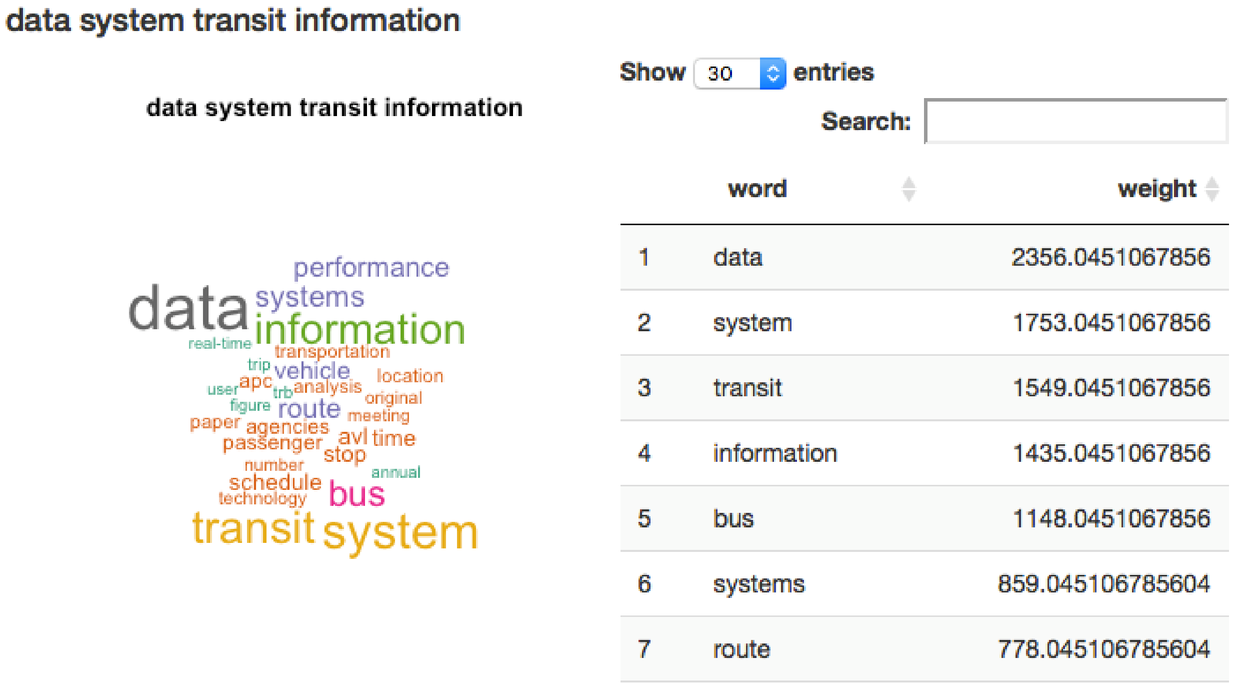
Once a topic is selected, use the numerical slider to choose how many words to view in the word cloud. 30 is recommended.


Number of Words in Cloud

30

Figure 19 Select Number of Words in Cloud

With these selections made, click “Update Table and Cloud” to view the topic. A word cloud will appear along with a data table that allows the user to view the words and their corresponding weights within the topic.



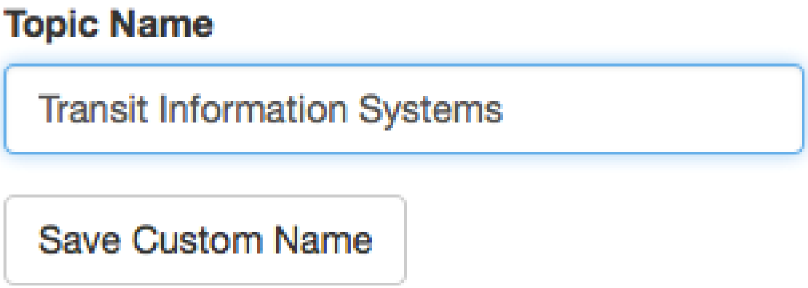
data system transit information

data system transit information

	word	weight
1	data	2356.0451067856
2	system	1753.0451067856
3	transit	1549.0451067856
4	information	1435.0451067856
5	bus	1148.0451067856
6	systems	859.045106785604
7	route	778.045106785604

Figure 20 Sample Word Cloud and Table for Topic

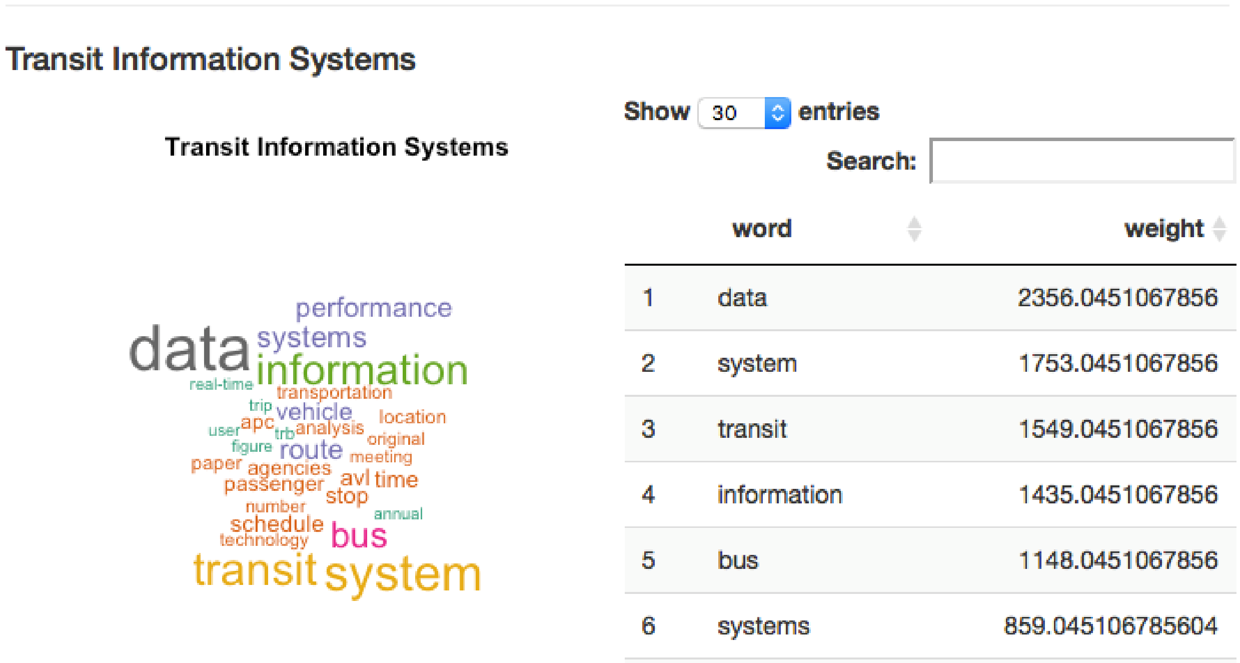
After viewing the word cloud and the topic, the user can rename the topic more appropriately. In the “Topic Name” box, replace the current name with a more appropriate one and click “Save Custom Name”. Remember, this will save the topic name to the project, but not to the save file on disk. (The user must click “Save Project” or “Save Project As” in the sidebar to do this.)



Topic Name: Transit Information Systems

Figure 21 Saving a Custom Topic Name

Once the custom topic name has been saved, clicking the “update Table and Cloud” will change the title of the word cloud.


Transit Information Systems

Transit Information Systems

Show 30 entries

Search:

	word	weight
1	data	2356.0451067856
2	system	1753.0451067856
3	transit	1549.0451067856
4	information	1435.0451067856
5	bus	1148.0451067856
6	systems	859.045106785604

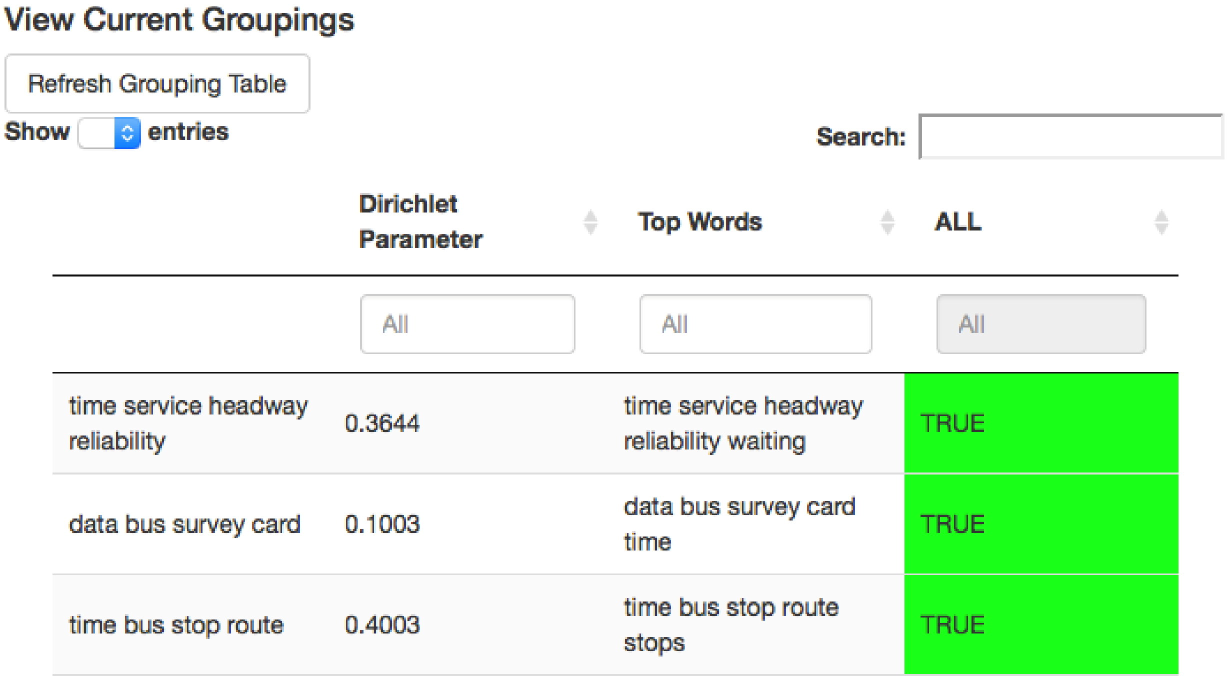
Figure 22 Sample Word Cloud and Table with Custom Topic Name

Repeat with all topics until satisfied with the topic names.

3.4. Group Topics

The group topics tab allows the user to group topics together in a logical way and overview all of the topics in the topic model. The topic groupings are used to exclude structure topics from the graph files, as desired. Eventually, they will be used in making the dynamic time series graphs.

To get an overview of the current topics and their groupings, click “Refresh Grouping Table”. If the user navigates to other tabs and changes topic names or other values, this button must be clicked again to refresh the table.



View Current Groupings

	Dirichlet Parameter	Top Words	ALL
	All	All	All
time service headway reliability	0.3644	time service headway reliability waiting	TRUE
data bus survey card	0.1003	data bus survey card time	TRUE
time bus stop route	0.4003	time bus stop route stops	TRUE

Figure 23 Sample Topic Grouping

There are three ways to edit topic groupings:

a. **Make a New Grouping**

To make a new grouping, click the “Make a New Grouping” button underneath “Select an Action”. Type the new groupings name into the name box. Then select topics from the drop down box to include in the grouping.

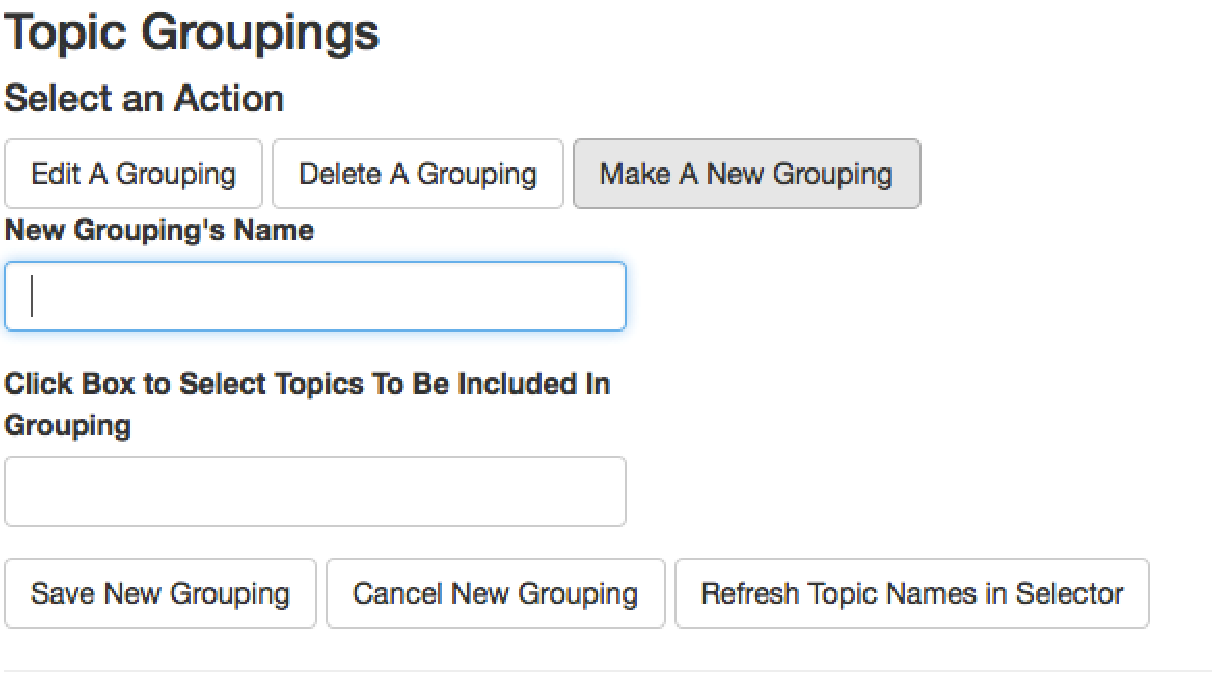


Figure 24 Make a New Topic Grouping

Once satisfied with the grouping, click “Save New Grouping” to save the grouping. The table overviewing the groupings will update.

Remember this saves it to the current Shiny Topic Model instance, but does not save it to the disk. The “Save Project” and “Save Project As” button must be used for this.

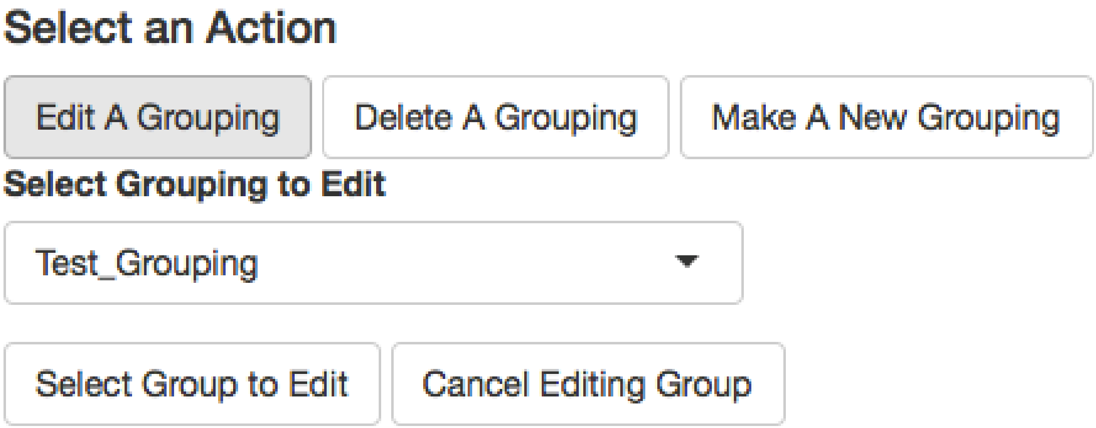
Click the “Cancel New Grouping” button to cancel making a new grouping.

If the topic names in the group selector are incorrect, click the “Refresh Topic Names in Selector” to refresh them. This will make no topics selected in the selector.

b. **Edit a Grouping**

To edit a grouping, the user must have created at least one grouping.

Click the “Edit A Grouping” button to make the grouping edit options appear. Use the drop down menu to select a grouping to edit. It will be blank if there are no groupings available to be edited.

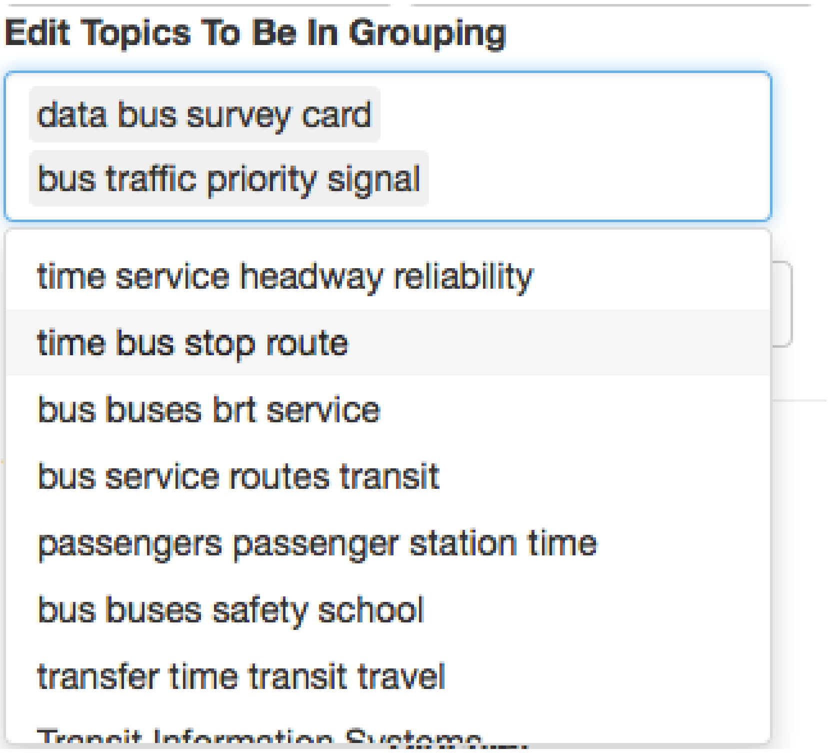


Test_Grouping

Figure 25 Select a Grouping to Edit

To edit the selected grouping click “Select Group to Edit”. To cancel editing a group, select “Cancel Editing Group”.

After selecting “Select Group to Edit”, a dropdown will appear with the included topics in the given grouping. Click to select the desired topics.


Edit Topics To Be In Grouping

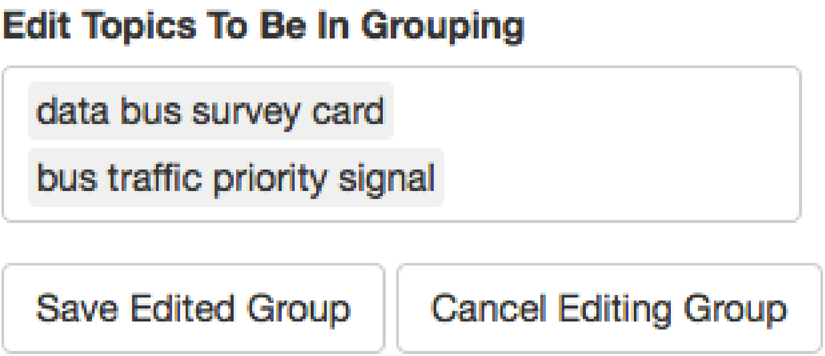
data bus survey card
bus traffic priority signal

time service headway reliability
time bus stop route
bus buses brt service
bus service routes transit
passengers passenger station time
bus buses safety school
transfer time transit travel
Transit Information Systems

Figure 26 Editing a Grouping

Once satisfied with the edited, selected topics click “Save Edited Group” to save the edited group. Click “Cancel Editing Group” to cancel editing the group.

Remember this saves it to the current Shiny Topic Model instance, but does not save it to the disk. The “Save Project” and “Save Project As” button must be used for this.


Edit Topics To Be In Grouping

data bus survey card

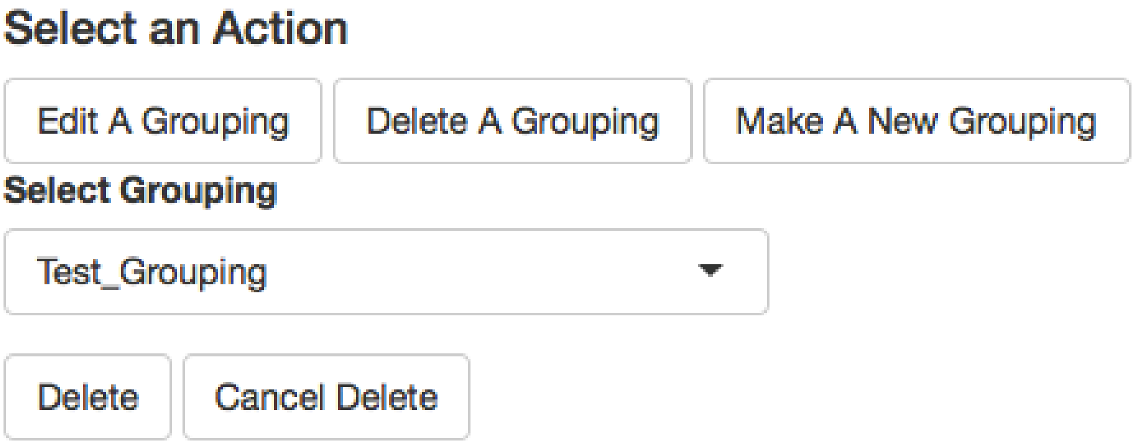
bus traffic priority signal

Figure 27 Saving Edited Group

c. **Delete a Grouping**

To delete a grouping the user must have already created a grouping. Click the “Delete A Grouping” button, and the delete a grouping option will appear.

In the drop down menu, select a group to delete. Click “Delete” to delete the grouping or “Cancel Delete” to cancel deleting a group.



Select Grouping

Test_Grouping

Figure 28 Delete a Grouping

On clicking the “Delete” button, the user is presented with confirmations options. Select the desired one to delete or cancel deleting the grouping.

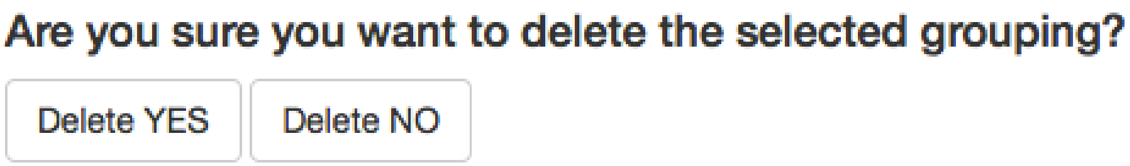

Are you sure you want to delete the selected grouping?

Figure 29 Delete Grouping Confirmation

3.5. View Data

The data tab allows the user to view and explore the document topic data directly.

Initially the user is only presented with a button, “Update Data Table”. Clicking the button updates the data table.

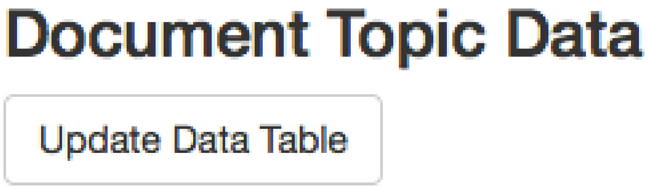
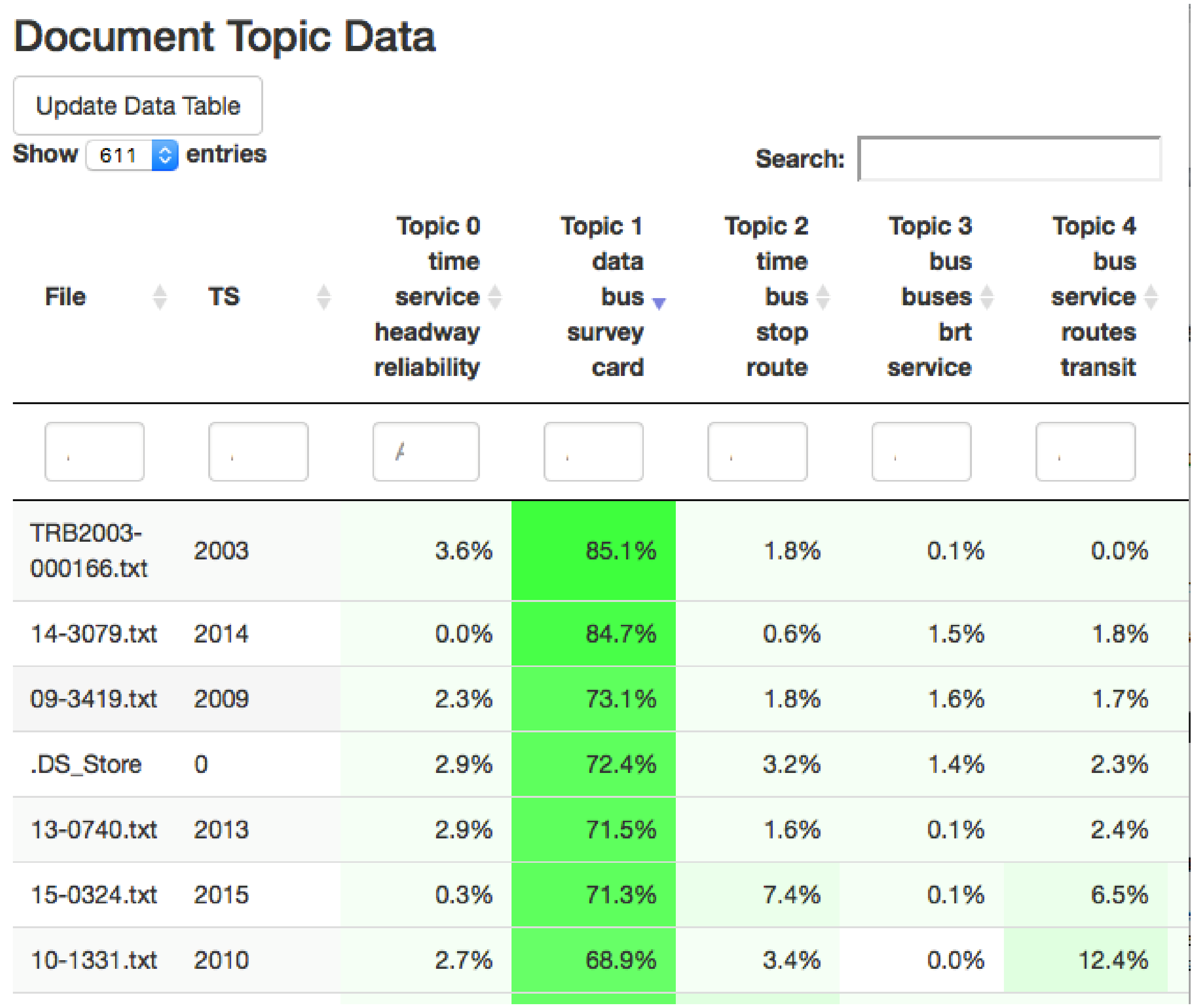

Document Topic Data

Figure 30 Update Data Table Button

Once the data table loads, the user can search for a document by name, sort documents by their proportion in given topics, and explore the data as desired. If the “Use Time series?” option was checked on the model tab and the files were in appropriate subfolders, the user can see which year the files were created in as well.



Document Topic Data

Update Data Table

Show 611 entries

Search:

		Topic 0	Topic 1	Topic 2	Topic 3	Topic 4
		time	data	time	bus	bus
File	TS	service	bus	bus	buses	service
		headway	survey	stop	brt	routes
		reliability	card	route	service	transit
TRB2003-000166.txt	2003	3.6%	85.1%	1.8%	0.1%	0.0%
14-3079.txt	2014	0.0%	84.7%	0.6%	1.5%	1.8%
09-3419.txt	2009	2.3%	73.1%	1.8%	1.6%	1.7%
.DS_Store	0	2.9%	72.4%	3.2%	1.4%	2.3%
13-0740.txt	2013	2.9%	71.5%	1.6%	0.1%	2.4%
15-0324.txt	2015	0.3%	71.3%	7.4%	0.1%	6.5%
10-1331.txt	2010	2.7%	68.9%	3.4%	0.0%	12.4%

Figure 31 Sample Topic Document Data Sorted By Topic 1 Proportion

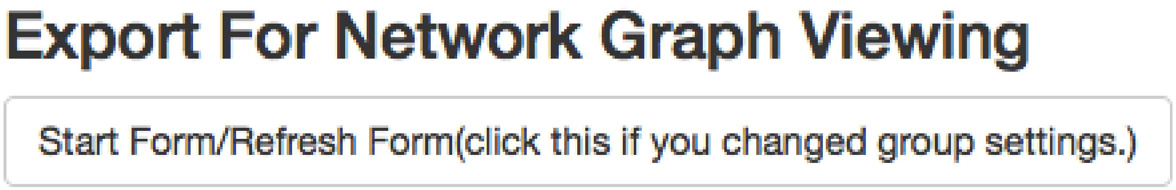
If the user changes any topic names, they will need to reselect the “Update Data Table” button to see the correct headers.

NOTE: FOR LARGE TOPIC MODELS, THE DATA TABLE CAN BE VERY, VERY SLOW. USING THE RAW R DATA OR THE EXCEL WORKBOOK WILL BE MORE EFFICIENT.

3.6. Create Graphs (GEXF)

The graph tab allows the user to create GEXF files for visualization within network graph software. These files are designed for Gephi, but other visualization software can likely be used.

To begin preparing for an export graph, the user should click the “Start Form/Refresh Form” button. Clicking this button collects the topic groupings and topic names on the other tabs, makes sure they are up to date, and allows the user to continue making a graph file.

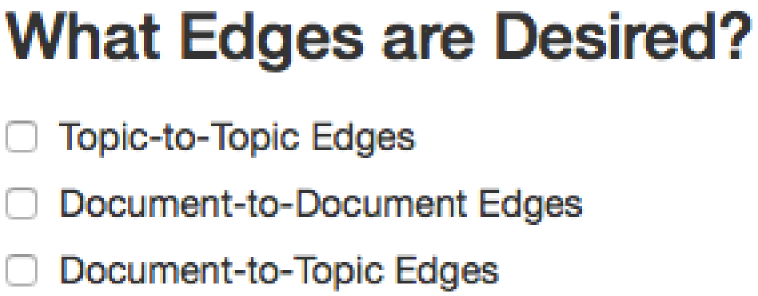


Export For Network Graph Viewing

Start Form/Refresh Form(click this if you changed group settings.)

Figure 32 Refresh Grouping Form Button

The user should then select the desired edge types to have within the resulting network graph file. One of these checkboxes must be selected; otherwise the user can select whichever ones they want. For best clustering visualizations, it is not recommended to use the Topic-To-Topic edges with any of the other options.



What Edges are Desired?

□ Topic-to-Topic Edges

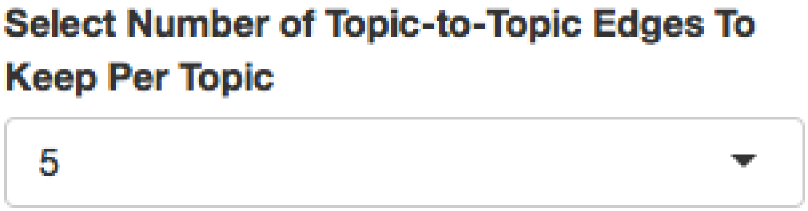
□ Document-to-Document Edges

□ Document-to-Topic Edges

Figure 33 Desired Edges Checkboxes

For each of the checkboxes selected, appropriate options appear below.

On selection of Topic-To-Topic edges, a drop down menu appears allowing the user to select the number of topic-to-topic edges to keep per topic. This is a method of normalizing the topic-to-topic network graph. Typically, the more dominant topics in a document have stronger connections to other topics purely as a result of their size, which overwhelms the non-dominant topics connections. Keeping only the top n topics per node allows the dominant connections of each topic to be visible.



Select Number of Topic-to-Topic Edges To Keep Per Topic

5

Figure 34 Selecting Number of Topic-to-Topic Edges

On selection of Document-to-Document Edges or Document-to-Topic Edges, a box appears allowing the user to select the target documents in the data set. This will change the node color and node size of the target documents within the network graph file, making them easier to pick out.

To select target documents, click in the box and scroll until the target documents are found. Alternatively, type the name of the file to search for the desired target.

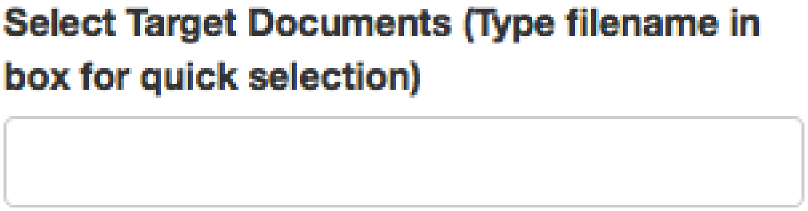

Select Target Documents (Type filename in box for quick selection)

Figure 35 Selecting Target Documents

On selection of any of the edges, a box appears allowing the user to select groupings of topics to exclude from the calculation of edge weights and from the network graph itself. This is useful for removing structural topics for the network graph or for creating a sub-graph of interest.

To select groupings to exclude, click in the box and select the groupings from the dropdown menu. Multiple groupings can be excluded, though the program will fail if the user tries to exclude all topics.

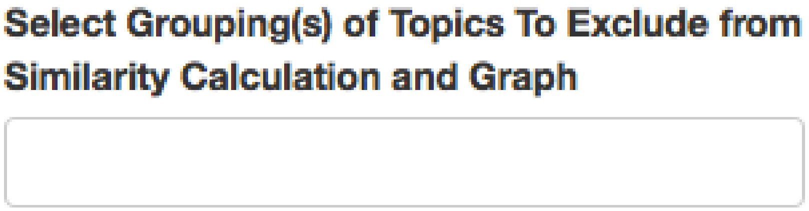
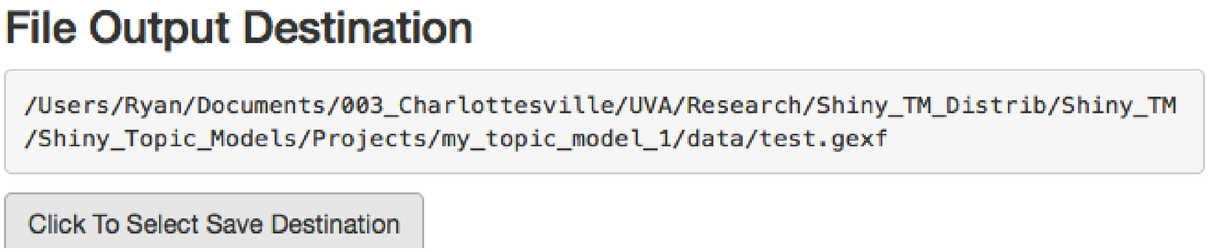

Select Grouping(s) of Topics To Exclude from Similarity Calculation and Graph

Figure 36 Excluding Topics Form Network Graph

Finally, the user needs to select a destination for the file output. Clicking the "Click to Select Save Destination" button allows the user to select a place on their computer for the gexf file. This should be a ".gexf" extension path. If the file already exists, the user will be prompted to overwrite or select an alternative location.



```
/Users/Ryan/Documents/003_Charlottesville/UVA/Research/Shiny_TM_Distrib/Shiny_TM
/Shiny_Topic_Models/Projects/my_topic_model_1/data/test.gexf
```

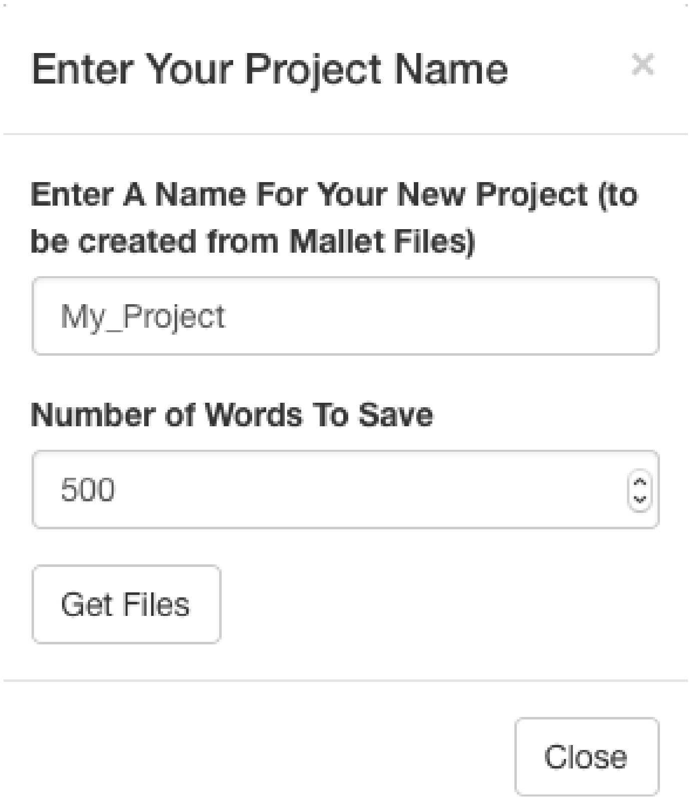
Figure 37 Setting Graph File Save Destination

Clicking the "Save Gexf File" button will write the Gexf file to disk. Depending on the size of the topic model and the graph settings, this process can take a long time so users should be patient and check the R Studio Console if they are worried if the computer has hung up.

3.7. Create Project From Existing Mallet Files

The user has the option to skip the settings and model tab, and create a new project from mallet output files made outside of Shiny Topic Models.

In the left side bar, selecting the "Create Project from Mallet" button opens a popup window.



Enter Your Project Name

Enter A Name For Your New Project (to be created from Mallet Files)

My_Project

Number of Words To Save

500

Figure 38 Building a Project from Mallet Files

After entering a name for your project and selecting a number of words to save for each topic (analogous to the setting in the setting tab), clicking the "Get Files" button opens a series of windows in which the user selects a place to save the file and the appropriate mallet files to use. First, the user is prompted to select a file path to save the project (give it a ".JSON" extension). Second, the user is prompted for the Topic Word Weights File. Third, the user is prompted for the Document Topics file. Finally the user is prompted for Topic Key file.

Once these are selected, Shiny Topic Models will build the project and save it the location specified. Then it will load it into the application and the remaining tabs can be used as if it was created within Shiny Topic Models.

3.8. Loading and Saving Prior Projects

Prior projects can be saved and loaded.

If no project is loaded, the left side bar has a button to "load project". Clicking this button opens a pop-up window where the user can select the ".json" file that contains the project.

If a project is loaded, the left side bar has buttons for "Save Project" and "Save Project As". Both of these buttons save the ".json" project file to the disk. The "Save Project" button saves it to the last place it was loaded, overwriting all prior data while the "Save Project As" allows the user to select a new location to save the ".json" file.

3.9. View Graphs with Gephi

The Gexf files exported by Shiny Topic Models can be opened with Gephi for easy visualization.

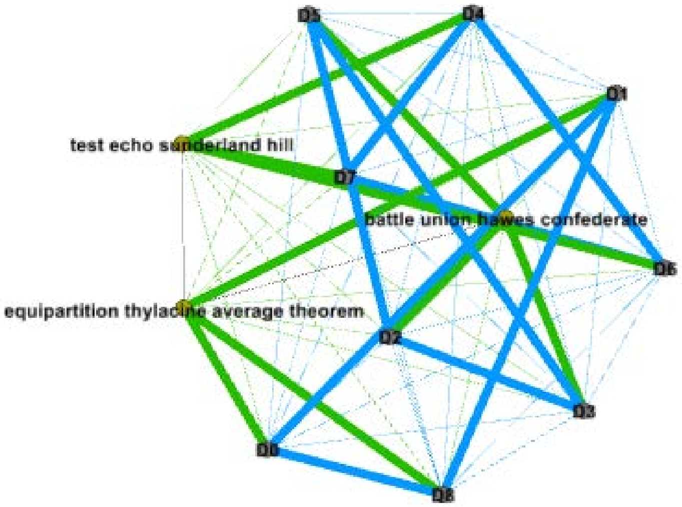

D5
D4
D1
test echo sunderland hill
D7
battle union hawes confederate
D6
equipartition thylacine average theorem
D2
D3
D0
D8

Figure 39 A network Graph in Gephi in the Fruchterman Reingold Layout

After opening Gephi, select file → open. Then select the Gexf file that was exported.

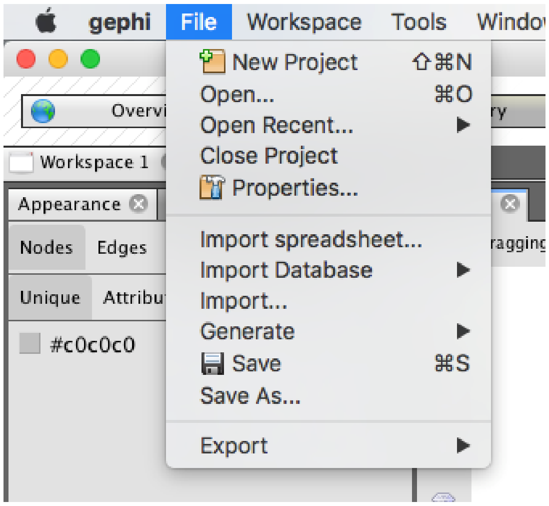
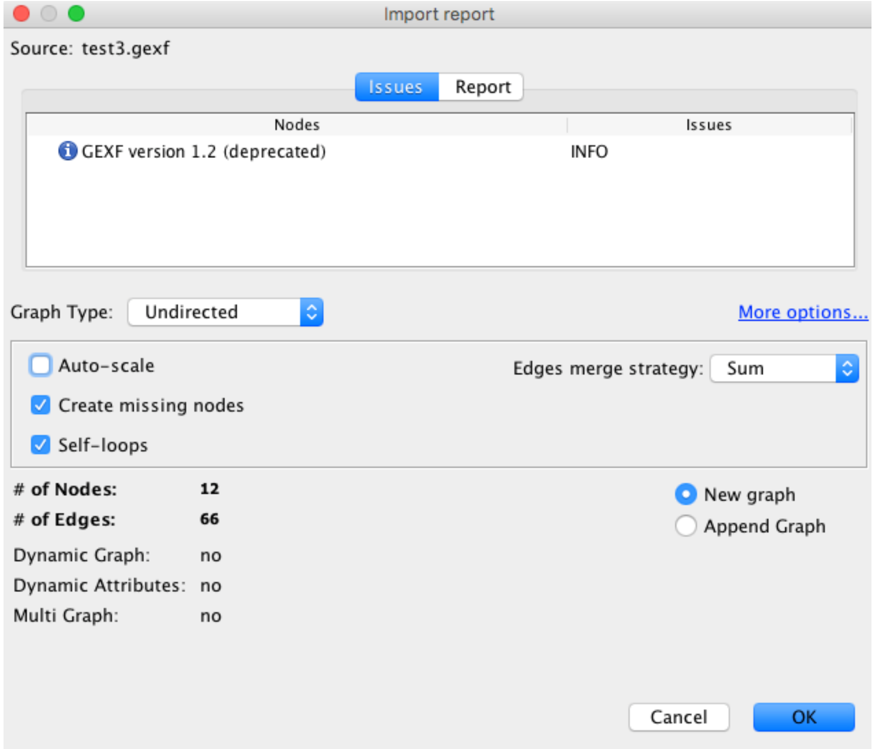


Figure 40 Opening a File in Gephi

Finally, click the "more options" link and unselect "auto-scale". (As of August 2016, this feature is bugged in Gephi and is on regardless of selection in the options. However, when loading the graphs you would want it unselected. This, when fixed, will load the nodes at the correct sizes.) Afterwards, click "OK" to load the graph file.



Import report

Source: test3.gexf

Issues | Report

Nodes	Issues
GEXF version 1.2 (deprecated)	INFO

Graph Type: Undirected

More options...

☐ Auto-scale

Edges merge strategy: Sum

☑ Create missing nodes

☑ Self-loops

# of Nodes:	**12**
# of Edges:	**66**
Dynamic Graph:	no
Dynamic Attributes:	no
Multi Graph:	no

● New graph

○ Append Graph

Figure 41 GEXF Import Screen with “Auto-Scale” Unselected

The network graph will appear in Gephi.

There are several things that a user can quickly to do gain information from these graphs:

a. **Label the nodes.**
First the user can put titles on the nodes of the network graph by click the large “T” at the bottom of the visualization window. The documents are named “D###” but the data laboratory button allows these user to identify which documents belong to which number.

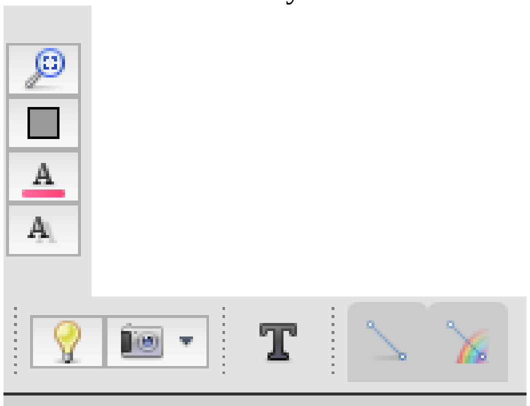
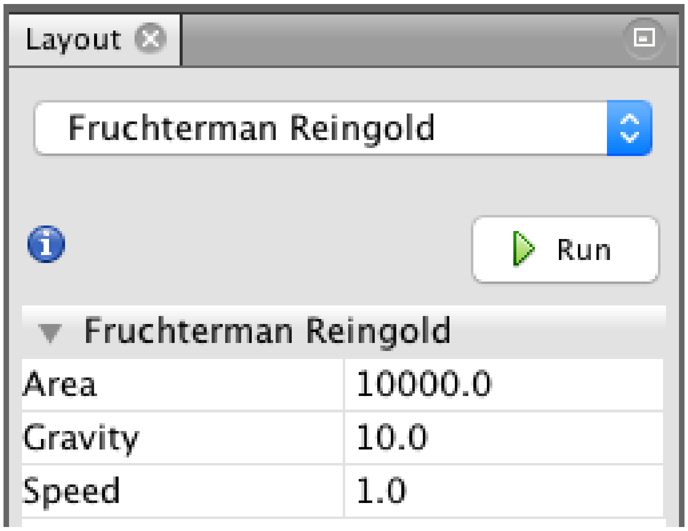


Figure 42 The black T shows the node names

b. **Visualize The Network.**
The network can be viewed in many differed ways by selecting options from the layout tab. After selecting a desired visualization from the drop down, click the “Run” button.



Area	10000.0
Gravity	10.0
Speed	1.0

Figure 43 Network Layouts in Gephi

There are several recommend visualizations:

Visualization	Description
Fruchterman Reingold	Equidistant circle clustering visualization
Force Atlas 2	Clustering
Open Ord	Clustering
Yifan Hu	Clustering
Label Adjust	Moves all nodes as little as possible so that all labels are visible, ideal after using other visualization

c. **Gather the Graph**
To gather the visualization click the blue magnifying glass in the bottom left corner of the window.

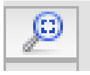
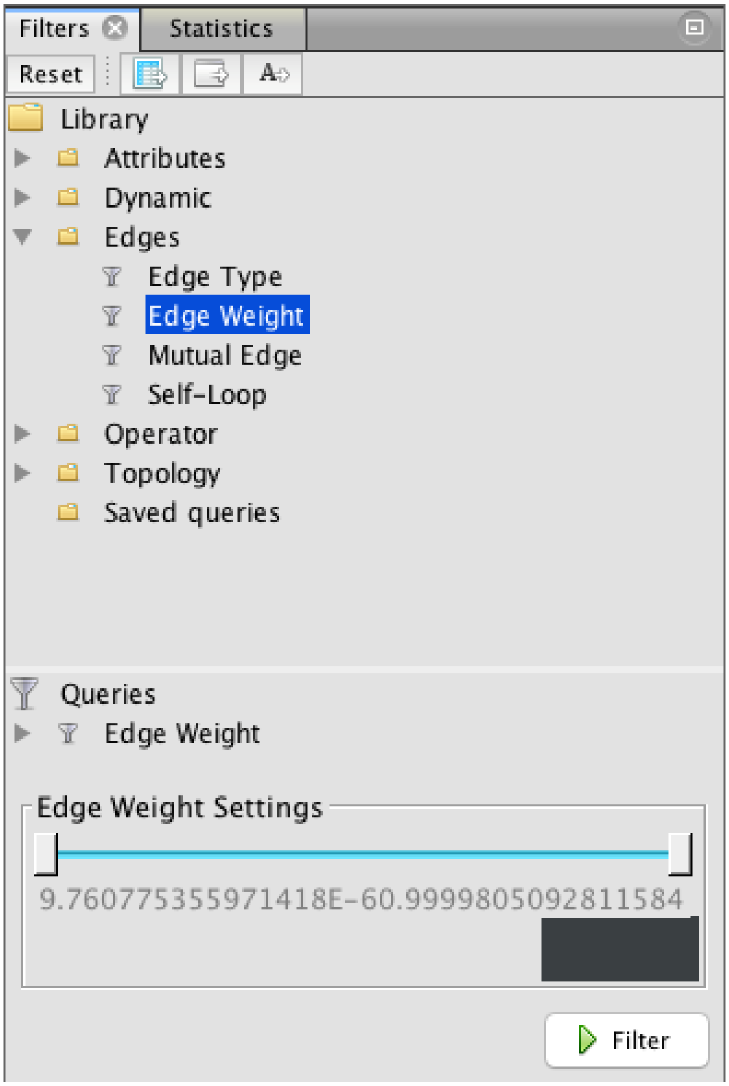


Figure 44 “Find Graph” Button in Gephi

d. **Filter Graph Edges by Weight**
The many edges of a Gephi graph can be filtered by edge weight. On the right sidebar, select the filters tab. underneath the edges menu, double click on “Edge Weight”. Drag the sliders from either end, then click filter.



9.760775355971418E-60.9999805092811584

Figure 45 Edge Weight Filter in Gephi

e. **Export Graph Files**
   Select the “Preview” button below the main menu.

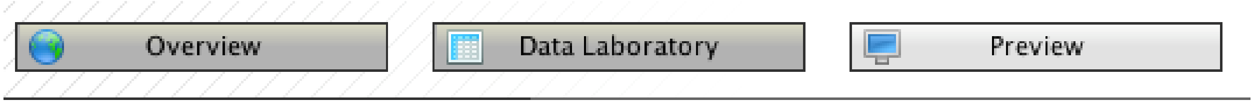


Figure 46 Main Buttons in Gephi; Preview is Far Right

After selecting the desired settings in the left side bar, click the “refresh” button to visualize the data and the “SVG/PDF/PNG” button to export the graph

4. Converting Files to TXT

At this point in time Shiny Topic Models does not contain functionality to convert pdf files to txt. Thankfully, many free, open-source, and effective solutions exist, which are listed here.

a. XPDF:
This is a command line or terminal tool that can be used to convert PDFs to text files.

Download Link: http://www.foolabs.com/xpdf/download.html

b. PyPDF2 Package in Python:
This is a python package that can convert PDF files to text files.

Download Link: https://pypi.python.org/pypi/PyPDF2/1.26.0
Documentation: http://pythonhosted.org/PyPDF2/

c. Apple Automator (Mac Only, resource inefficient.)
Mac users can create an apple automator workflow that converts PDFs to text. This is great for those who feel uncomfortable on the command line or with python, but this method is much more resource intensive and slower.

Tutorial: https://www.engadget.com/2013/02/11/mac-101-use-automater-to-extract-text-from-pdfs/

5. Future Improvements

Shiny Topic Models has potential for strong growth. Immediate opportunities for improvements include:

- Handling of file paths with spaces in mallet (rebuild mallet using custom java library)
- Improved window focus on windows for getting file paths
- Inclusion of LDA Viz package for topic model visualization
- Inclusion of ability to view time series graphs of topics through time
- Inclusion of XPDF functionality to strip text from PDF files
- Cleaning and prettying of layout and UI
- Inclusion of alternative ways of measuring similarity between topics and documents (currently just using dot product).

6. A Word About Server Hosting Functionality

Shiny is an application server at heart, not a standalone application development structure. Topic modeling is not ideal for use on a server due to the tremendous about of data used and the amount of bandwidth it would consume (gigabytes of text files and gigabytes of raw output data).

Shiny topic models was built to be used with one computer functioning as both the client and server; it should be run locally on one computer. Shiny offers no natural way to get file paths, so TCLTK was used as a replacement for many of the IO infrastructure and the user warning messages.

Shiny Topic Models could be recreated as a server application, but the file IO would need to be rebuilt.

7. Known Limitations and Bugs

- Mallet cannot handle file paths with spaces. Any directory or extra stop words file passed to mallet cannot have a space anywhere in the file path. This is an inherent problem in Mallet (not shiny), and cannot be easily fixed without rebuilding the mallet infrastructure. Current work around is to store your files somewhere without spaces.
    - E.g. "/Users/LastName FirstName/files/mytextfiles/" would fail as there is a space between "lastname" and "firstname"
    - "/Users/LastName_FirstName/Files/mytextfiles/" would work as there is an underscore instead of a space
- On Windows the TCLTK windows for getting save paths popup behind the focus windows of R Shiny and R.

8. Selected References

[1] David M Blei, Andrew Y. Ng, and Michael I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, pp. 993-1022, January 2003.

[2] David Blei, "Probabilistic Topic Models," *Communications of the ACM*, vol. 55, no. 4, April 2012.

[3] Stuart Geman and Donald Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence* , vol. PAMI-6, no. 6, pp. 721-741, November 1984.

[4] Andrew Kachites McCallum. (2002) MALLET: A Machine Learning for Language Toolkit. [Online]. HYPERLINK "http://mallet.cs.umass.edu" http://mallet.cs.umass.edu

[5] Ryan C Boyer, William T Scherer, and Michael C Smith, "Trends Over Two Decades of Transportation Research: A Machine Learning Approach," *96th TRB Annual Meeting Compendium of Papers (Submitted)*, 2017.

Appendix D – Acronyms

DST	Dimensions of Systems Thinking
GEXF	Graph Exchange XML Format
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IO	Input/Output
JRE	Java Runtime Environment
JSON	Java Script Object Notation
LDA	Latent Dirichlet Allocation
PDF	Portable Document Format
PNG	Portable Network Graphics
SDK	Software Development Kit
SVG	Scalable Vector Graphics
TRAC-MTRY	TRADOC Analysis Center, Monterey
UI	User Interface
UVA	University of Virginia
XML	eXtensible Markup Language